

AD-A259 764



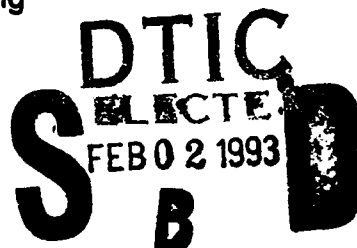
WL-TR-92-3024

**VIBRATION CONTROL
EXPERIMENTS ON A
12-METER CANTILEVER
TRUSS STRUCTURE**



Ümit Özgüner, Stephen Yurkovich, Peter Dix,
Layne Lenning, Anne Bruner, and Ken Cheung

The Ohio State University
1960 Kenny Road
Columbus, Ohio 43210



August 1992

Final Report for Period November 1987 - November 1991

Approved for public release; distribution is unlimited.

**FLIGHT DYNAMICS DIRECTORATE
WRIGHT LABORATORY
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6553**

93-01839



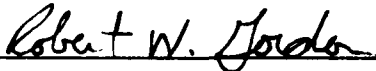
11618

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely Government-related procurement, the United States Government incurs no responsibility or any obligation whatsoever. The fact that the Government may have formulated or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication, or otherwise in any manner construed, as licensing the holder, or any other person or corporation; or as conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nationals.

This technical report has been reviewed and is approved for publication.



ROBERT W GORDON
Project Engineer



STEPHEN R WHITEHOUSE, Maj, USAF
Technical Manager, Vibration Group



JEROME PEARSON
Chief, Structural Dynamics Branch

If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify WL/FIBGC, WPAFB, OH 45433-6553 to help us maintain a current mailing list.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specified document.

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
<small>Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.</small>				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE August 1992		3. REPORT TYPE AND DATES COVERED Final - Nov. 1987 - Nov. 1991
4. TITLE AND SUBTITLE Vibration Control Experiments on a 12-Meter Cantilever Truss Structure			5. FUNDING NUMBERS C F33615-87-C-3257 PE 62201F PR 2401 TA 04 WU 41	
6. AUTHOR(S) Unit Ozguner, Stephen Yurkovich, Peter Dix, Layne Lenning, Anne Bruner, Ken Cheung				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Ohio State University Research Foundation 1960 Kenny Rd. Columbus, Ohio 43210			8. PERFORMING ORGANIZATION REPORT NUMBER RF 766464/720289	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Robert Gordon (513-255-5236) Flight Dynamics Directorate Wright Laboratory Wright Patterson Air Force Base, Ohio 45433-6553			10. SPONSORING / MONITORING AGENCY REPORT NUMBER WL-TR-92-3024	
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION / AVAILABILITY STATEMENT APPROVED FOR PUBLIC RELEASE; DISTRIBUTION IS UNLIMITED			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) Control objectives for large flexible space structures such as reorientation maneuvers, vibration suppression, fine pointing, and precision shape control require reliable and robust control algorithms. The modal characteristics of such structures including large order, low frequency with low damping, and closely spaced modes of vibration demand complexities in the control design which go beyond traditional methodologies and present difficult challenges. The Large Space Structures Technology Program at the Flight Dynamics Directorate of the Wright Laboratory was begun as an in-house experimental research program in the dynamics and control of flexible space structures. The 12-Meter Truss Active Control Experiment has been conducted as part of the Large Space Structures Technology Program to address modeling and active vibration damping issues on a representative laboratory-scale model of future large flexible space structures. Our objectives, as part of the 12-Meter Truss Active Control Experiment, have been to identify several controller design techniques for active vibration damping, to examine these techniques via simulation studies, and to implement a number of the designs on the 12-meter truss. This report describes experiment, modeling, and the design and testing of several control approaches on the 12-meter truss.				
14. SUBJECT TERMS Large Space Structures, Active Control, Experiments			15. NUMBER OF PAGES 116	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT SAR	

FOREWORD

This report is submitted to the Flight Dynamics Directorate, Wright Laboratory, Wright-Patterson Air Force Base, Ohio, in partial fulfillment of data requirements under contract F33615-87-C-3257, "Controller Design Approaches and Evaluations." The Air Force project engineer is Robert W. Gordon, WL/FIBGC.

Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

Contents

1	INTRODUCTION	1
1.1	Overview	1
1.2	Large Flexible Structures	1
1.2.1	Background and Related Work: Theoretical	1
1.2.2	Background and Related Work: Experimental	4
1.3	Description of the Experimental Facility	6
2	ANALYSIS AND DESIGN OF CONTROLLERS FOR FLEXIBLE STRUCTURES	8
2.1	Modeling Issues	8
2.1.1	Partial Differential Equation Model	8
2.1.2	Finite Element Model	13
2.2	Model Order Reduction	14
2.2.1	Transformation from Nodal Coordinates to Modal Coordinates	14
2.2.2	Balanced Reduction	15
2.3	Controller Design Approaches	18
2.3.1	Linear Quadratic Gaussian (LQG)	18
2.3.2	Decentralized Control	20
2.3.3	Centralized and Decentralized Optimal Output Feedback . . .	22
2.3.4	Decentralized Frequency Shaping (DFS)	24
2.3.5	Overlapping Decompositions	25
2.3.6	Isolated Boundary Loading and Controlled Component Synthesis (CCS)	30
2.3.7	Maximum Entropy/Optimal Projection (MEOP)	35
2.3.8	H_∞ Design Approach	38
3	EXPERIMENTAL HARDWARE CONFIGURATION	42
3.1	Truss Structure, Controllers and Actuators	42

3.2	Real-time Control System	51
4	SOFTWARE FOR CONTROL, SIMULATION, AND DATA AC-	
	QUISITION	52
4.1	Goals of the Software	52
4.2	Control/Simulation Program	52
4.2.1	Software Overview	52
4.2.2	Computer Compatibility	54
4.2.3	Memory Usage and Allocation	55
4.2.4	Sampling Rate	55
4.2.5	Output of the Software	57
4.2.6	Addition of New Controllers	57
4.3	Post-Processor Routines	58
4.3.1	Capabilities and Structure	58
5	SYSTEM MODELING AND CONTROLLER DESIGN FOR AFWAL	
	TRUSS	61
5.1	System Modeling	61
5.1.1	Modeling of Control Forces and Sensor Signals	61
5.1.2	Transformation from Nodal Coordinates to Modal Coordinates	63
5.1.3	Model Order Reduction	64
5.1.4	Effects of Discretization and Sampling Delay	64
5.2	Controller Design	65
5.2.1	Performance Requirement—Tip Pointing Error	65
5.2.2	Controller Design Techniques	66
5.2.3	Output Feedback and Actuator Damping	71
6	SIMULATION AND EXPERIMENTAL RESULTS	72
6.1	Noise Levels of Velocity Signals	72
6.2	Disturbance Generation	73
6.3	Bode Plots	76

6.4	RMS Velocities	84
6.5	Modal Damping	84
6.6	Time Responses	87
6.7	H_{∞} Gap Metric Design Results	87
7	CONCLUSIONS AND RECOMMENDATIONS	93
A	Sample Run of Control/Simulation Program	104

List of Figures

2.1	Inserting a Controller	34
2.2	Standard Feedback Configuration	39
3.1	A 12-Meter Vertical Cantilever Truss Structure	43
3.2	12-Meter Truss Experiment in Vibration Test Facility	44
3.3	Momentum Exchange Actuator	45
3.4	12-Meter Truss Actuator	46
3.5	Actuator Locations on Truss	47
3.6	12-Meter Truss 1/2 Station Actuators	48
3.7	12-Meter Truss Tip Station Actuators	49
3.8	System Overview	50
4.1	Speed of a Matrix/Vector Multiplication using Optima 3's Array Processor	56
4.2	Sample Run of Post-Processor	59
5.1	Model of Proof Mass Actuator	62
5.2	Sensor and Actuator Numbering for Controller Designs	63
6.1	Bode Plot of Shaker with Voltage Drive	73
6.2	Bode Plot of Shaker with Current Drive	74
6.3	Bandwidth Limited Random Disturbance Signal	75
6.4	Comparison of Experimental and Simulation Open Loop Frequency Responses, Velocity #1, Shaker 1-lb RMS Random	77
6.5	Comparison of Experimental and Simulation DOFB Frequency Responses, Velocity #1, Shaker 1-lb RMS Random	77
6.6	Comparison of Experimental and Simulation COFB Frequency Responses, Velocity #1, Shaker 1-lb RMS Random	78
6.7	Comparison of Experimental and Simulation Overlapping Decomposition Frequency Responses, Velocity #1, Shaker 1 lb RMS Random	78
6.8	Comparison of Experimental and Simulation Controlled Component Synthesis Frequency Responses, Velocity #1, Shaker 1-lb RMS Random	79

6.9	Comparison of Experimental Frequency Responses for Several Controllers, Velocity #1, Shaker 1-lb RMS Random	79
6.10	Comparison of Experimental and Simulation Open-Loop LOS Frequency Responses, LOS x , Shaker 1-lb RMS Random	80
6.11	Comparison of Experimental and Simulation DOFB LOS Frequency Responses, LOS x , Shaker 1-lb RMS Random	81
6.12	Comparison of Experimental and Simulation COFB LOS Frequency Responses, LOS x , Shaker 1-lb RMS Random	81
6.13	Comparison of Experimental and Simulation Overlapping Decomposition LOS Frequency Responses, LOS x , Shaker 1 lb RMS Random . .	82
6.14	Comparison of Experimental and Simulation Controlled Component Synthesis LOS Frequency Responses, LOS x , Shaker 1-lb RMS Random	82
6.15	Comparison of Simulation LOS Frequency Responses for Several Controllers, LOS x , Shaker 1-lb RMS Random	83
6.16	Typical Sinusoidal Disturbance Signal, 0.2-lb RMS 1.75 Hz	87
6.17	Initial Response - Open Loop, Velocity #1, 0.2-lb RMS 1.75-Hz Source 1	88
6.18	Initial Response - DOFB #2, Velocity #1, 0.2-lb RMS 1.75-Hz Source 1	88
6.19	Decay Response - Open Loop, Velocity #1, 0.2-lb RMS 1.75-Hz Source 1	89
6.20	Decay Response - DOFB #2, Velocity #1, 0.2-lb RMS 1.75-Hz Source 1	89
6.21	Comparison of Experimental Open Loop and Closed Loop Frequency Responses for Constant Weight H_∞ Design, $k = 0.5$, Velocity #1, Shaker 1-lb RMS Random	91
6.22	Comparison of Experimental Open Loop and Closed Loop Frequency Responses for Second-Order Weight H_∞ Design, $k = \frac{0.50(0.1s+53)^2}{s^2+26.5s+2809}$, Velocity #1, Shaker 1-lb RMS Random	92
6.23	Comparison of Experimental Open Loop and Closed Loop Frequency Responses for Second-Order Weight H_∞ Design, $k = \frac{0.55(0.1s+52)^2}{s^2+26s+2704}$, Velocity #1, Shaker 1-lb RMS Random	92

List of Tables

5.1	Structural Modes and Dampings — Open Loop, 10% Actuator Damping	64
5.2	Structural Modes and Dampings — Open Loop, 50% Actuator Damping	64
6.1	Typical RMS Noise Levels of Velocity Signals (inch/s)	72
6.2	Experimental RMS Sensor Velocities and Actuator Power for Random Disturbance, 1-lb RMS	85
6.3	Simulated RMS Sensor Velocities and Actuator Power for Random Disturbance, 1-lb RMS	85
6.4	Simulation Closed Loop Damping Ratios (in %) — Minimized For Pointing Requirement, 10% Actuator Damping	86
6.5	Simulation Closed Loop Damping Ratios — Minimized For Pointing Requirement, 50% Actuator Damping	86
6.6	Damping Ratios of Structural Modes, b_{opt} , and Weighted Gaps for H_{∞} Designs	90

1. INTRODUCTION

The Flight Dynamics Directorate at Wright Laboratory (WL/FIBGC) has long had an interest in the dynamic and control aspects of large space structures [1, 2, 3]. In 1985, the Large Space Structures Technology Program (LSSTP) was initiated. One of the goals of this program has been to develop a laboratory in which structures dynamically similar to future LSS can be studied. The 12-meter truss described here is part of this effort, involving both active and passive methods of vibration control. WL has developed the design and model of the structure and the setup of the experiment. The Ohio State University, Control Research Laboratory researchers have designed the active controllers, implemented the software for control, and performed the closed-loop testing.

1.1 Overview

The present report summarizes the total effort in this project, including the theory, design, simulation studies, and implementation results of active vibration control of a 12 meter cantilevered truss structure.

1.2 Large Flexible Structures

There are typically three areas which compose Large Space Structure (LSS) control:

- Vibration damping
- Pointing (slewing)
- Static shape deformation adjustment

Each of the above requires effort in modeling, control design, actuation and sensor development, and resolution of practical application details.

The present effort addresses the first area—that of vibration damping.

1.2.1 Background and Related Work: Theoretical

Many approaches have been advocated from the theoretical viewpoint for vibration suppression in flexible structures. Most LSS control schemes for vibration damping share in the goals for robustness and reduced-order models. During the last ten years,

a number of approaches have been suggested and have been tested to some extent. Among the early techniques one can list:

- Basic colocated velocity feedback
- HAC/LAC
- Positivity based feedback
- Independent Modal Control

More recently, other (mostly state-space based) techniques have also been analyzed. The following should be highlighted:

- MEOP
- LQG/LTR
- Variable Structure (sliding-mode control)
- Various Adaptive Control Approaches
- H_∞ -based techniques
- Decentralized versions of the above, developed using different techniques

The approaches which have been truly developed to some extent for real systems are a small subset of the above. In the present research we only had an opportunity to consider some of these avenues.

The first issue one has to face in designing controllers for flexible structures is that of modeling and model reduction. In fact, in a number of approaches, modeling and model reduction are coupled into the controller design.

In general, a Finite Element Model (FEM) is developed, tested, and provided to the control designer. At this point, one can simply truncate the dynamics associated with higher frequencies. For beam-like structures, this may be a valid approach because various criteria based on control authority, energy content, etc., indicate retention of the mode ordering sequence. For more complex structures, however, one will have to utilize an approach based on balanced reduction or some "cost" comparison-based scheme. When considering decentralized control we used an approach that our Group has utilized previously at JPL, *decentralized balanced realization*.

Two other approaches were also evaluated: Maximum Entropy/Optimal Projection (MEOP) and an H_∞ -based method using the GAP Metric. In both of these cases,

model reduction and modeling errors are part of the controller design. Modeling errors and effects of "discarded" modes are also considered to an extent by frequency shaping, which was used with decentralized control.

Although many feedback controllers can be designed for decentralized systems, so far the most popular has been based on Linear Quadratic cost criteria. Here we also have considered decentralized Linear Quadratic control and utilized a locally developed software package named *DOLORES*. One aspect considered in this context was frequency weighting.

Early versions of frequency weighting may be attributed to the *High Authority Controller/ Low Authority Controller* approach introduced by Aubrun [4] and covered in some detail in the ACROSS Three (Phase I) Final Report. As analyzed there, Low-Authority Controls provide limited (broadband) modal damping and may be synthesized using perturbation techniques. It has been shown that for systems restricted to colocated rate sensing and decentralized feedback loops, stability is guaranteed over the system bandwidth. High-Authority Control provides arbitrary damping ratios in low frequency modes by exploiting knowledge of system dynamics and is generally synthesized as a multi-input, multi-output (centralized) control. The spillover phenomena in the HAC is handled by utilizing frequency shaping techniques such as presented in [5]. We have reported preliminary results in designing the High Authority Controller also assuming decentralized implementation in [6] where a decentralized quadratic regulator with frequency weighting was analyzed. As covered by the software package *DOLORES*, various options can be analyzed and one can be selected for implementation. Another of our decentralized control efforts is based on the overlapping decomposition approach—a very promising idea which has only recently been developed. Overlapping decompositions were first utilized by Ikeda, Šiljak, and co-workers [7, 8, 9, 10]. Özgüner and his students [11] and Young [12, 13] have extended the approach further for decentralized controller design.

In general, decentralized control implies that each control channel is assigned the task of regulation of a portion of the state space. In the context of flexible structures, each controller used in active vibration damping is associated with a subset of vibrational modes. This type of decomposition can be performed in an overlapping manner so that some portion of the state space is assigned to more than one channel [14], or similarly, one set of modes is assigned to more than one controller.

Once the local models are obtained through the expansion of overlapping decompositions, feedback controllers are designed separately. This leads to uncoupled feedback controllers in the expanded state-space, which are then "contracted" to the original state-space for implementation.

Further details of the above mentioned approaches may be found in the Chapter 2 of this report.

1.2.2 Background and Related Work: Experimental

Experimental studies on active vibration damping on laboratory scale structures have been continuing for a number of years. Recent surveys of experiments and facilities are provided in [15, 16]. There are several experimental structures and facilities on which LSS research is conducted.

At the Charles Stark Draper Laboratory (CSDL), the Observation/Control Spillover Experiment investigated active vibration damping of a 60-inch cantilevered fixed-free aluminum beam [17]. Later experiments have been performed on the Flexible Satellite Slew Testbed which consists of a rigid central hub with 4 equally-spaced flexible arms extending radially to a diameter of 9 feet. The entire apparatus is suspended by a vertical-axis air bearing [18].

At the NASA Langley Research Center (LaRC) in Hampton, Virginia, several experiments have been conducted. In early experiments, active vibration damping of a 12-ft-long free-free uniform aluminum beam [19] and a 7-ft by 10-ft flexible grid [20] has been considered. Vibration control experiments have also been conducted on the 20-meter, 18-bay mini-mast testbed in which the vertical truss is cantilevered at the base [21]. The Spacecraft Control Laboratory Experiment (SCOLE) investigates the control of a rigid platform with a 10-ft beam connected to the bottom of the platform. At the opposite end of the beam is an offset reflector with a diameter of 40 inches [22]. More recent work is being conducted on structures in the Space Structures Research Laboratory (SSRL) at NASA LaRC [23, 24]. The Dynamic Scale Model Test (DSMT) Structure is a 500-inch meriform truss oriented horizontally and cantilevered at one end. The CSI Evolutionary Model (CEM) is a 52-ft truss structure which is oriented horizontally and entirely suspended by cables. This structure has a rib antenna substructure attached to one end.

At the Jet Propulsion Laboratory (JPL) of the California Institute of Technology in Pasadena, California, the USAF Astronautics Laboratory and NASA jointly sponsor experimental work in conjunction with the Large Flexible Structure Test Facility. Early vibration suppression and shape control experiments were performed on a 16-ft vertical pinned-free flexible beam [25]. Also, in the Piezoelectric Active Member Experiment, piezoceramic material was used for sensing and actuation of a 12.5-in horizontal, clamped-free aluminum beam [26]. Later vibration suppression work was performed on the Flexible Structure Testbed which consists of a rigid hub structure with 12 equally spaced horizontal ribs projecting out for a total diameter of 19 feet [27]. More recent experimental structures include the Modified Astromast, the Precision Truss, and a 13-bay horizontal truss which can either be cantilevered or suspended at its midspan [28].

At the USAF Astronautics Laboratory (AFAL), the Large Space Systems Laboratory

has performed experiments on the Grid Test Article which is a 5-ft by 5-ft aluminum grid oriented vertically and cantilevered from the top [29]. Also at AFAL, the Advanced Space Structure Technology Research Experiment (ASTREX) Facility has been developed. This structure consists of an 5.5-meter diameter primary mirror support structure with three 5.2-meter graphite epoxy tubes forming a tripod shape over the primary mirror. The tripod structure is the secondary mirror support metering truss. The entire apparatus is suspended by a spherical air bearing with a diameter of 48 cm. The air bearing is sits atop a 4.5-meter pedestal allowing the structure to rotate about all three axes [30, 31].

At the NASA Marshall Space Flight Center (MSFC) in Huntsville, Alabama, experiments are conducted at the Large Space Structure Ground Test Facility (LSS-GTF). The Active Control Technique Evaluation for Spacecraft (ACES) structure is a 13-meter astromast with an offset antenna with a diameter of 3 meters attached at the bottom. The structure is oriented vertically and is cantilevered at the top. A laser, mirrors, and an optical detector attached to the antenna are used for pointing control experiments. Other types of experiments performed on this structure include vibration suppression, system identification, and fault detection and isolation studies [32, 33, 34].

Other experimental facilities and structures exist at Martin Marietta Denver Aerospace, Lockheed Missile and Space Company (LMSC), Harris Corporation, TRW, The Ohio State University (OSU), the Virginia Polytechnic Institute and State University (VPI&SU), the Massachusetts Institute of Technology (MIT), the State University of New York at Buffalo (SUNY-Buffalo), and Texas A&M University.

The Ohio State Control Research Laboratory has been involved in such -based studies for about 6 years, on structures both in-house and at U. S. Government institutions. The experimental efforts in validating control design approaches for vibration damping of flexible structures have been very successful. The techniques considered and further developed as necessary, ranged from simple Linear Quadratic feedback to variable structure controllers in a decentralized configuration.

At the OSU Control Research Laboratory, we have performed experiments on vibration damping on a free-free beam with proof-mass actuation [35, 36], slewing for single-link [36, 37, 38] and two-link structures [39] and an experiment involving multiple mirror orientations on a slewing flexible link [39].

In conjunction with NASA LaRC, we addressed analysis and design methodologies for control and parameter estimation of large flexible space structures. The focus of the work was also on development and validation of technology, and subsequent experimentation, that relates to the NASA LaRC Spacecraft Control Laboratory Experiment (SCOLE) [40, 41]. The primary objective of the research was threefold: first, to assess the performance of various methods, primarily from an adaptive control

viewpoint, in simulation tests and actual experimentation; and second, to determine the effectiveness of various techniques of parameter estimation for use in control applications. Finally, simplified minimum-time slewing for flexible structures was also studied.

At JPL, we investigated various decentralized control approaches primarily related to the overlapping decomposition approach. The main thrust of the project was verification of the decentralized control approaches on the JPL/AFAL Flexible Structure Testbed. Also, some adaptive control and fault detection schemes were investigated [42].

1.3 Description of the Experimental Facility

The structure considered is a 12-meter truss constructed from square aluminum tubing, with cross members of Lexan plastic. It is composed of four sections—each consisting of 4 truss bays. These sections are bolted together to form the truss with a total of 16 bays. For closed-loop control, the truss is oriented in the vertical cantilever position [43, 44].

Active damping is achieved through eight momentum exchange proof-mass actuators. These devices are essentially linear DC motors—each supplied by its own current driver and capable of supplying up to 1 pound force from 2 Hz to 100 Hz. Pairs of actuators are mounted perpendicular to each other at the center of both the one-half and three-quarter stations on the truss, thus affecting only the bending modes. There are four actuators at the tip, two in each direction, mounted off center. In this way, both torsion and bending can be affected. Sensors for the control system are accelerometers, which are mounted on the actuator housings. The acceleration signals are integrated to provide velocity signals. A point light source is located at the tip. An optical sensor can record the position of the light source to provide information on the displacement of the tip.

The truss is excited via two alternative sources. The first is an additional proof mass actuator, identical to the control actuators. The second alternative is a large shaker, mounted to the wall at the tip of the truss. The shaker applies a force up to 10 lb through a light connecting arm.

The control algorithms are implemented digitally using a custom control computer manufactured by Systolic Systems, Inc. The system consists of the control computer and a Sun 3/50 graphics workstation that serves as a development system. External control is accomplished through a 12-channel, 16-bit, A/D and D/A card. The control programs are written in C programming language, compiled on the Sun workstation, and downloaded to the control computer. This system is capable of implementing

rather complex control algorithms at frequencies needed for the truss.

Data acquisition and analysis can be performed either by uploading data to the Sun host or through an OnoSokki dual channel spectrum analyzer. Since the interface between the control computer and the workstation is very slow, the spectrum analyzer is frequently used to minimize the time required to collect data for each experiment.

2. ANALYSIS AND DESIGN OF CONTROLLERS FOR FLEXIBLE STRUCTURES

Partial differential equation (PDE) models and finite element models (FEMs) are often used to model flexible structures. For controller design, the FEM is often transformed from nodal coordinates to *modal coordinates* by simultaneously diagonalizing the mass and stiffness matrices. The model, put into state space form, can then be reduced using balanced model reduction techniques. Controllers are subsequently designed for the reduced-order model.

2.1 Modeling Issues

2.1.1 Partial Differential Equation Model

The first and probably most obvious problem is modeling the structural dynamics. Large space structures are infinite dimensional. Using the notation of [45], they may be described, in general, by a system of partial differential equations (PDE):

$$m(x)\ddot{u}(x,t) + D_0\dot{u}(x,t) + A_0u(x,t) = F(x,t)$$

where $u(x,t)$ represents the instantaneous displacement of the structure off of the equilibrium position; $m(x)$ is the mass density; $D_0\dot{u}$ is the damping term; A_0u is the internal restoring force term; and $F(x,t)$ is the external force distribution.

A_0 is a positive semi-definite differential operator on the infinite dimensional Hilbert space of square integrable functions (with the usual inner product and associated norm) defined on (the interval containing) the structure, $\Omega : H_0 = L^2(\Omega)$. The operator is usually assumed to have a discrete spectrum such that it satisfies the eigenproblem

$$A_0\phi_k = \omega_k^2\phi_k$$

where ω_k represents the modal frequencies and ϕ_k represents the mode shapes.

The external force distribution can be decomposed into control forces and external disturbance forces (including nonlinearities):

$$F(x,t) = F_c(x,t) + F_d(x,t)$$

The control forces are due to M actuators and may be described by:

$$F_c = B_0 f.$$

where f represents the actuator amplitudes and B_0 represents the actuator influence functions (which, in general, allow for both point and distributed actuation).

Measurements from P position or velocity sensors (including integrated accelerometer measurements) can be described as

$$y = C_0 u + E_0 \dot{u}$$

where C_0 represents the influence functions of the position sensors and E_0 represents the influence functions of the velocity sensors (which also allow for both point and distributed measurement).

State Space Formulation from the PDE Model

Since the truss can basically be treated as a cantilevered beam, we shall generate a state space representation of the system dynamics from the Euler-Bernoulli PDE model. Consider,

$$\frac{EI_y}{\rho A} \frac{\partial^4 u(x, t)}{\partial x^4} + \frac{\partial^2 u(x, t)}{\partial t^2} = 0 \quad (2.1)$$

where

ρ = Density of Beam

A = Cross Sectional Area of Beam

E = Young's Modulus

I = Area Moment of Inertia of Beam

ω = Natural Frequency (rad/sec)

The solution can now be obtained by employing separation of variables:

$$u(x, t) = r_z(x) q_z(t) \quad (2.2)$$

Substitution of Equation (2.2) into Equation (2.1), yields a time dependent and a position dependent equation.

$$\frac{d^2 q_z(t)}{dt^2} + \omega_z^2 q_z(t) = 0 \quad (2.3)$$

$$\frac{d^4 r_z(x)}{dx^4} - \frac{\rho A}{EI_y} \omega_z^2 r_z(x) = 0 \quad (2.4)$$

A general solution of the position dependent equation is chosen to be

$$r_z(x) = A_z \sin \beta_z x + B_z \cos \beta_z x + C_z \sinh \beta_z x + D_z \cosh \beta_z x \quad (2.5)$$

Applying boundary conditions will allow us to solve for the coefficients and the vibrational frequencies.

The beam can now be forced with a set of point force actuators applied at specified locations. By introducing forces, $F_a(t)$, acting transversely on the beam for the $z - x$ bending, at actuator locations x_a ($a = 1, 2, \dots, N_a$), and equating these forces with Equation (2.1) results in

$$\frac{EI}{\rho A} \frac{\partial^4 u(x, t)}{\partial x^4} + \frac{\partial^2 u(x, t)}{\partial t^2} = \frac{1}{\rho A} \sum_{a=1}^{N_a} \delta(x - x_a) F_a(t) \quad (2.6)$$

where

$$u(x, t) = \sum_{i=1}^{\infty} r_{zi}(x) q_{zi}(t) \quad (2.7)$$

Substituting Equation (2.7) into Equation (2.6) gives

$$\frac{EI_y}{\rho A} \sum_{i=1}^{\infty} \left(\frac{d^4 r_{zi}(x)}{dx^4} \right) a_{zi}(t) + \sum_{i=1}^{\infty} r_{zi}(x) \ddot{q}_{zi}(t) = \frac{1}{\rho A} \sum_{a=1}^{N_a} \delta(x - x_a) F_a(t) \quad (2.8)$$

From Equation (2.4),

$$\frac{d^4 r_z(x)}{dx^4} = \beta_z^4 r_z(x) \quad (2.9)$$

where

$$\beta_z^4 = \left(\frac{\rho A}{EI_y} \right) \omega_z^2$$

Upon substitution of Equation (2.9) into Equation (2.8), one obtains

$$\sum_{i=1}^{\infty} (\omega_{zi}^2 r_{zi}(x) q_{zi}(t) + r_{zi}(x) \ddot{q}_{zi}(t)) = \frac{1}{\rho A} \sum_{a=1}^{N_a} \delta(x - x_a) F_a(t) \quad (2.10)$$

For the $r_{zi}(x)$ orthonormal with respect to

$$\int_0^L r_{zi}(x) r_{zj}(x) dx = \begin{cases} 0, & i \neq j \\ 1, & i = j \end{cases} \quad (2.11)$$

Equation (2.11) is multiplied by $r_{zj}(x)$ and integrated over $\int_0^L (\cdot) dx$ to obtain

$$\begin{aligned} \int_0^L \left\{ \sum_{i=1}^{\infty} r_{zi}(x) r_{zj}(x) (\ddot{q}_{zi}(t) + \omega_{zi}^2 q_{zi}(t)) \right\} dx = \\ \frac{1}{\rho A} \int_0^L \left\{ \sum_{a=1}^{N_a} r_{zj}(x) \delta(x - x_a) F_a(t) \right\} dx, \quad j = 1, \dots, \infty \end{aligned} \quad (2.12)$$

which reduces to

$$\ddot{q}_{zj}(t) + \omega_{zj}^2 q_{zj}(t) = \frac{1}{\rho A} \sum_{a=1}^{N_a} r_{zj}(x_a) F_a(t) \quad (2.13)$$

Expanding Equation (2.13) for the first n-modes,

$$\begin{aligned} \ddot{q}_{z1} + \omega_{z1}^2 q_{z1} &= \frac{1}{\rho A} r_{z1}(x_1) F_1(t) + r_{z1}(x_2) F_2(t) + \dots + r_{z1}(x_{N_a}) F_{N_a}(t) \\ \ddot{q}_{z2} + \omega_{z2}^2 q_{z2} &= \frac{1}{\rho A} r_{z2}(x_1) F_1(t) + r_{z2}(x_2) F_2(t) + \dots + r_{z2}(x_{N_a}) F_{N_a}(t) \\ &\vdots \\ \ddot{q}_{zn} + \omega_{zn}^2 q_{zn} &= \frac{1}{\rho A} r_{zn}(x_1) F_1(t) + r_{zn}(x_2) F_2(t) + \dots + r_{zn}(x_{N_a}) F_{N_a}(t) \end{aligned} \quad (2.14)$$

and dropping the z subscript, Equation (2.14) can be more generally represented by

$$\ddot{\mathbf{q}} + \begin{bmatrix} \ddots & & \\ & \omega_n^2 & \\ & & \ddots \end{bmatrix} \mathbf{q} = \frac{\Phi^T \mathbf{F}}{\rho A} \quad (2.15)$$

Note that all mode shapes are to be evaluated at the point of actuation. The state-space model is therefore

$$\dot{\Omega} = A\Omega + \begin{bmatrix} 0 \\ \Phi^T \end{bmatrix} F \quad (2.16)$$

where

$$\Omega \triangleq [q \dot{q}]^T$$

$$A \triangleq \begin{bmatrix} 0 & I \\ -\omega_n^2 & 0 \end{bmatrix}$$

Inclusion of Actuators and Sensors into an Expanded State-Space Model

The sensor dynamics are implicitly included in the overall state-space model by assuming that the output of the colocated sensors is velocity, and not acceleration. The sensor scale factors, in volts/m/sec, are the only values taken into account, since the sensors had flat frequency responses over the range of interest. Only the output C , D -matrices need to be adjusted to reflect this fact (i.e., C has nonzero elements corresponding to velocity states, D is the null matrix). Therefore, it is the actuator dynamics that must be combined with the truss dynamics.

Define the beam/truss dynamics to be the following:

$$\dot{X}_Y = A_Y X_Y + B_Y F_Y \quad (2.17a)$$

$$\dot{X}_X = A_X X_X + B_X F_X \quad (2.17b)$$

$$\dot{X}_T = A_T X_T + B_T F_T \quad (2.17c)$$

the actuators to have dynamics,

$$\dot{X}_i = A_i X_i + B_i V_i + G_{Yi} X_Y + G_{Ti} X_T; \quad i = y \text{ direction} \quad (2.18a)$$

$$\dot{X}_i = A_i X_i + B_i V_i + G_{Xi} X_X + G_{Ti} X_T; \quad i = x \text{ direction} \quad (2.18b)$$

and the forces acting in each degree of freedom,

$$F_Y = C_i x_i + D_i V_i \quad (2.19a)$$

$$F_X = C_i x_i + D_i V_i \quad (2.19b)$$

$$F_T = C_i x_i + D_i V_i \quad (2.19c)$$

where

$$A_Y, A_X, A_T = \left[\begin{array}{c|c} 0 & I \\ \hline -\omega^2 & -2\xi\omega \end{array} \right]$$

$$B_Y, B_X, B_T = \begin{bmatrix} 0 \\ \psi \end{bmatrix}$$

$$A_i = \begin{bmatrix} 0 & 1 \\ -k_i & -d_i \end{bmatrix} \quad i = 1, 2, 3, 4$$

$$B_i = \begin{bmatrix} 0 \\ G_i \end{bmatrix} \quad i = 1, 2, 3, 4$$

$$C_i = m_p[-k_i \quad -d_i] \quad i = 1, 2, 3, 4$$

$$D_i = m_p G_i \quad i = 1, 2, 3, 4$$

where

$$\begin{aligned} \text{State Vector} &= [X_Y \ X_X \ X_T \ X_1 \ \dots \ X_N]^T \\ \text{Input Vector} &= [V_1 \ \dots \ V_N]^T \\ \text{Output Vector} &= [\dot{Y} \ \dot{Z} \ \dot{\theta}]^T \end{aligned}$$

where

$$V_i = \text{Voltage input to power amplifier - } i$$

2.1.2 Finite Element Model

Note that the description discussed in the previous section is for infinite-dimensional distributed parameter systems. Obviously, it would be impractical (if not impossible) to design controllers for LSS using an infinite-dimensional model. Therefore, a finite-dimensional model is desired. Perhaps the most popular method is finite element modeling (FEM) where the resulting model is of the form

$$M\ddot{z} + D\dot{z} + Kz = Q \quad (2.20)$$

where M , D , and K are the mass, damping, and stiffness matrices, respectively; z is the nodal coordinate vector (displacements and velocities in each direction, at each node); and Q is the generalized force vector. Note that D is usually positive semi-definite and symmetric representing the internal structural damping, but can also contain a skew symmetric component representing gyroscopic damping due to any on-board rotors or constant spin rate of the whole LSS. M is a positive definite, symmetric

matrix, and K is a positive semi-definite, symmetric matrix. These matrices can be generated from any of several existing FEM software packages such as NASTRAN, SPAR, STARDYN, etc. Whatever the case, the structural dynamicist obtains a FEM of the form above.

2.2 Model Order Reduction

2.2.1 Transformation from Nodal Coordinates to Modal Coordinates

One basic approach to order reduction is through consideration of the modal distribution. In order to use this approach for the FEM, the model must be converted from nodal coordinates to modal coordinates by simultaneously diagonalizing the mass and stiffness matrices. These matrices may be diagonalized in a variety of ways resulting in differently balanced relations between the inputs to states and states to outputs transformations. The balanced realization obtained here results in $C_{pt} = B_{pt}^T$, where C_{pt} describes the relation between the nodal state vector and a velocity sensor, and B_{pt} describes the relation between a point force input at the same location and the nodal states. This provides some symmetry in the model and makes the model more numerically stable for simulations.

The state vector is defined to be

$$x = \begin{bmatrix} x_m \\ \dot{x}_m \end{bmatrix} \quad (2.21)$$

where x_m represents the modal displacements and \dot{x}_m represents the modal velocities. To diagonalize the system, the matrix Φ is used to define the unitary transformation $z = \Phi x_m$ such that

$$\Phi^T M \Phi = I \quad (2.22)$$

and

$$\Phi^T K \Phi = \Omega^2 = \text{diag}(\omega_1^2, \dots, \omega_n^2) \quad (2.23)$$

where Ω represents the modal frequencies and n is the number of modes in the model.

This matrix Φ can be obtained using singular value decomposition (SVD) methods. First, find diagonal Ω_1 and unitary U_1 ($U_1^{-1} = U_1^T$) such that

$$\Omega_1 = U_1^T M U_1. \quad (2.24)$$

Let $U_2 = \sqrt{\Omega_1^{-1}}$ and $\hat{K} = U_2^T U_1^T K U_1 U_2$ such that \hat{K} is symmetric and positive semidefinite. Then, find diagonal Ω_2 and unitary U_3 such that

$$\Omega_2 = U_3^T \hat{K} U_3. \quad (2.25)$$

By defining $\Phi = U_1 U_2 U_3$, it is evident that (2.22) and (2.23) hold. Note that Ω^2 is merely Ω_2 . Now Equation (2.20) can be transformed to modal coordinates by substituting $z = \Phi x_m$ and premultiplying by Φ^T

$$\Phi^T M \Phi \ddot{x}_m + \Phi^T D \Phi \dot{x}_m + \Phi^T K \Phi x_m = \Phi^T B_q u \quad (2.26)$$

where B_q is the force distribution matrix. Now, Equation (2.26) can be rewritten in state space form using Equation (2.21)

$$\begin{bmatrix} \dot{x}_m \\ \ddot{x}_m \end{bmatrix} = \begin{bmatrix} 0 & I \\ -\Omega^2 & -\Phi^T D \Phi \end{bmatrix} \begin{bmatrix} x_m \\ \dot{x}_m \end{bmatrix} + \begin{bmatrix} 0 \\ \Phi^T B_q \end{bmatrix} u = Ax + Bu \quad (2.27)$$

The output matrix for velocities at the sensor locations can be written

$$y = Cx = \begin{bmatrix} 0 & C_q \Phi \end{bmatrix} \begin{bmatrix} x_m \\ \dot{x}_m \end{bmatrix} \quad (2.28)$$

where C_q is the sensor distribution matrix. The displacements at the actuator locations are,

$$y = C_d x = \begin{bmatrix} C_q \Phi & 0 \end{bmatrix} \begin{bmatrix} x_m \\ \dot{x}_m \end{bmatrix} \quad (2.29)$$

Thus, the structure is represented by a finite-dimensional model in modal coordinates.

2.2.2 Balanced Reduction

The modal transformation described in the last section results in a system equal in order to the FEM model. This model has too many modes to be used for controller design, and the higher order modes are certainly inaccurate. In modal coordinates, modes may be neglected simply by deleting the rows and columns of the state space representation which represent these modes. Modes were deleted according to their controllability and/or observability. Since the actuators and sensors are colocated, the controllability and observability for structural modes are closely related.

For the structural modes (as opposed to actuator modes), the controllability and observability are very strongly linked to the modal frequency—lower frequency modes are most important. In fact, for a white noise disturbance at most points on the truss, the first few modes very accurately describe the motion of the truss.

The decision of which modes to retain from each subsystem can be made by considering the grammians [46]. The basic idea is to transform a system so that its controllability and observability grammians are equal and diagonal (the system is *internally balanced*). For a system in this form, the singular values of the grammians are contained in a diagonal matrix with the values in descending order. With the controllability and observability grammians equal, we can then make meaningful statements about how controllable and observable a given mode is, and delete those modes from the model that are weakly controllable (and thus weakly observable).

In general, the controllability grammian W_c is the solution of the algebraic Lyapunov equation:

$$AW_c + W_c A^T + BB^T = 0 \quad (2.30)$$

while the observability grammian W_o satisfies the equation

$$A^T W_o + W_o A + C^T C = 0 \quad (2.31)$$

The block diagonal form of a system in modal coordinates (see equation (2.52a) for example) is exploited to give closed form solutions for W_c and W_o . For W_c , the 2×2 blocks W_{ij} are a solution of

$$A_i W_{ij} + W_{ij} A_j^T + B_i B_j^T = 0 \quad (2.32)$$

with A_i, B_i in modal coordinate form, and the closed form solution is

$$W_{ij} = \begin{pmatrix} 2\omega_i \omega_j (\zeta_j \omega_i + \zeta_i \omega_j) & \omega_j (\omega_j^2 - \omega_i^2) \\ -\omega_i (\omega_j^2 - \omega_i^2) & 2\omega_i \omega_j (\zeta_i \omega_i + \zeta_j \omega_j) \end{pmatrix} \beta_{ij} / d_{ij} \quad (2.33)$$

where $\beta_{ij} = b_i^T b_j$, d_{ij} is a measure of how closely correlated modes i and j are, and ω_i, ζ_i represents the modal frequency and damping of the i th mode. For the observability grammian W_o , Equation (2.32) takes a form similar to Equation (2.31) and β_{ij} in Equation (2.33) is replaced with $\gamma_{ij} = c_i^T c_j$.

For the case where $p \geq m$ (more sensors than actuators), it can be shown that the system is orthogonally symmetric if we can find U such that $C = UB^T P$ and P is $\text{diag}(1, -1, \dots, 1, -1)$ which satisfies $A^T = PAP$. The cross-grammian is now defined as W_{co} which satisfies

$$AW_{co} + W_{co}A^T + BU^TC = 0 \quad (2.34)$$

There are several relations of W_{co} that make it extremely useful for balancing applications:

$$\begin{aligned} W_{co}^2 &= W_c W_o \\ \text{and } W_{co} &= W_c P = P W_o \end{aligned} \quad (2.35)$$

Thus the equation above for W_{ij} works for all three grammians with appropriate changes in the β_{ij} term, where for W_{co} , $\beta_{ij} = \gamma_{ij}$.

Now we consider the balancing problem for a system in modal coordinates. The state transformation T to take the system described in state space as $\{A, B, C\}$ to balanced form $\{T^{-1}AT, T^{-1}B, CT\}$, with equal and diagonal grammians

$$T^{-1}W_c T^{-T} = T^T W_o T = \Sigma = \text{diag}(\sigma_i) \quad (2.36)$$

and with $\sigma_1 \geq \sigma_2 \geq \dots \geq 0$, consists of right eigenvectors of $W_c W_o$, since

$$\Sigma^2 = (T^{-1}W_c T^{-T}) (T^T W_o T) = T^{-1} (W_c W_o) T \quad (2.37)$$

For orthogonally symmetric systems, $W_{co}^2 = W_c W_o$ implies that all balancing information is in the eigenstructure of the cross-grammian:

$$T^{-1}W_{co}T = \Lambda \quad (2.38)$$

$$\Lambda^2 = \Sigma^2, \quad (2.39)$$

so $\Lambda = \text{diag}(\pm\sigma_i)$ with $\Sigma = \text{abs}(\Lambda)$.

The above results can be used for model reduction by choosing the most dominant modes by observing large magnitude changes of the singular values on the diagonal. The dominant balanced subsystem has state-space representation

$$\{\hat{A}, \hat{B}, \hat{C}\} = \left\{ (I, 0)T^{-1}AT \begin{pmatrix} I \\ 0 \end{pmatrix}, (I, 0)T^{-1}B, CT \begin{pmatrix} I \\ 0 \end{pmatrix} \right\} \quad (2.40)$$

where columns of $T \begin{pmatrix} I \\ 0 \end{pmatrix} = T_1$ are right eigenvectors of W_{co} corresponding to the largest eigenvalues of Λ_1 .

Actuator modes are slightly more complicated. When the model is transformed to modal coordinates, *the actuator modes do not correspond to single actuators*, but rather to a linear combination of all actuators.

2.3 Controller Design Approaches

This project presented the opportunity to test a wide variety of control techniques. These were all based on the Linear Quadratic Regulator (LQR). Linear quadratic regulator (LQR) design methods are common and convenient controller design techniques [47, 48]. The cost function minimized in these techniques is typically

$$J = \frac{1}{2} \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (2.41)$$

or some variation thereof. Controllers are often designed using the pointing requirements. The inputs are usually weighted equally, thus R is diagonal.

One particular method of interest is linear optimal output feedback where only the output is available for feedback [49]. Often in a LQR design, the outputs of a system are weighted rather than the states. The resulting cost criterion is of the form

$$J = \frac{1}{2} \int_0^{\infty} (y^T Q y + u^T R u) dt \quad (2.42)$$

where u and y are the system input and output vectors, respectively. Q and R are positive semi-definite and positive definite matrices, respectively, of appropriate dimensions. If Q is positive definite and the system is observable, this cost criterion is a Lyapunov function and the resulting controller asymptotically stabilizes the system.

All of these design techniques have been discussed thoroughly in other sources; therefore, only a short description will be given here.

2.3.1 Linear Quadratic Gaussian (LQG)

In this approach, it is assumed that the system equations are

$$\dot{x} = Ax + Bu + w \quad (2.43)$$

$$y = Cx + v \quad (2.44)$$

where $w \sim N(0, W)$ and $v \sim N(0, V)$ are independent white Gaussian processes. The cost criterion which is to be minimized is

$$J = \lim_{\tau \rightarrow \infty} \frac{1}{\tau} E \left\{ \int_0^\tau (x^T Q x + u^T R u) dt \right\} \quad (2.45)$$

This approach uses the standard full state feedback with a Kalman filter to estimate the modal states [50]. The Kalman filter is of the form

$$\dot{\hat{x}} = A\hat{x} + Bu + L(y - C\hat{x}) \quad (2.46)$$

where \hat{x} is the estimated state vector. The gain L is found by solving the Algebraic Riccati equation,

$$PA^T + AP + W - PC^T V^{-1} CP = 0 \quad (2.47)$$

where W is the process noise covariance matrix, V is the measurement noise covariance matrix and $L = PC^T V^{-1}$.

The state feedback was computed to minimize the cost function by solving a similar Riccati equation,

$$SA + A^T S - SBR^{-1}B^T S + Q = 0 \quad (2.48)$$

The feedback is given by $u = K\hat{x} = -R^{-1}B^T S\hat{x}$.

Linear Quadratic Gaussian with Loop Transfer Recovery (LQG/LTR)

The LQG/LTR approach [51, 52, 11] combines the frequency domain and state space techniques for a minimum phase system and is very similar to the LQG design discussed above. The fundamental idea in LTR design is to recover a target feedback loop (TFL) which satisfies some design performance specification with a suitable asymptotic design. An example of a TFL is a LQR design. If system states are not available and they have to be estimated, some of the robustness properties of the LQR design will be lost. LTR provides a remedy for this. The theory states that a filter (not a Kalman filter) can be designed such that the overall design—controller plus filter—will asymptotically behave as the controller design (LQR).

The primary difference between the LTR technique and the LQG technique is the filter design equation

$$PA^T + AP + BB^T - PC^T CP = 0 \quad (2.49)$$

The theory states that as the design parameter $q \rightarrow \infty$, the system feedback loop approaches the TFL asymptotically.

2.3.2 Decentralized Control

In the control of Large Flexible Space Structures, the notion of decentralization plays an important role. This notion is concerned with imposing information flow constraints on the various controllers of the system; controllers of the system are only allowed to measure certain outputs of the system and control through corresponding inputs. The decentralization constraint enters into large scale systems because it may be impractical or even impossible to communicate signals from one controller to another. Moreover, decentralization may be required by the control designer to achieve reliability and a degree of redundancy, and it may impose a structure to the control implementation by relegating control authority to separate channels.

A large scale system is called *decentralized* when either due to physical measurement/actuation constraints, or by design, there exists a one to one correspondence between sets of inputs and outputs. If we assume that the large scale system is linear, the state space model is,

$$\begin{aligned}\dot{x} &= Ax + \sum_{i=1}^N B_i u_i \\ y_i &= C_i x + D_i u_i \quad i = 1, \dots, N\end{aligned}\tag{2.50}$$

where the index i denotes an input/output channel.

During the last 10 years, a number of techniques have been considered for designing decentralized controllers. A number of optimal control approaches have been pursued by Davison [53, 54], Özgüner [55, 56], and others [57]. These basically define a Linear Quadratic (LQ) framework and may utilize various avenues for solving for the "optimal" feedback gains. Servo-compensators for tracking and disturbance rejection, implementation constraints and frequency weighting may be included.

Recently, adaptive control techniques have also been used for decentralized systems. Applications of model-reference adaptive control have been made to robotic systems and flexible structures. Indirect adaptive control has also been considered on a flexible structure.

Decentralization is particularly relevant in Large Flexible Space Structures for various reasons:

- The sensor/actuator locations on the structure have a decentralized nature which makes it difficult or even impossible to incorporate a centralized control architecture.
- Information transfer will be costly for complex space systems, so that the designer may wish to *impose* decentralized (local) feedback.

- On-line computations will be too time consuming for some structures so simpler decentralized strategies will be necessary.

A model for the large flexible structure to be controlled under a decentralized information structure is the following:

$$\begin{aligned} M\ddot{q} + Kq &= \sum_{i=1}^N B_i u_i \\ y_i &= C_{p_i} q + C_{v_i} \dot{q} \quad i = 1, \dots, N \end{aligned} \quad (2.51)$$

where $u_i \in \mathbb{R}^{m_i}$ is the vector of control inputs available at the i^{th} channel, $y_i \in \mathbb{R}^{r_i}$ is the vector of measurements available at the i^{th} channel, and $q \in \mathbb{R}^n$ is the nodal displacement vector. M and K are the inertia and stiffness matrices, respectively. Such a formulation is general enough to include the cases of position measurements only (through C_{p_i}) or velocity measurements only (through C_{v_i}), as well as the case of distributed control with noncollocated sensing. Note that in all of the modeling and control philosophies mentioned above, noncollocated feedback control schemes may also be used. That is, decentralized control does not necessarily imply colocated feedback, but rather the *assignment* of one or more sets of sensors to correspond to one or more sets of actuators. The primary claim with the formulation (2.51) is that certain measurements correspond to certain control inputs for purposes of feedback control. This decentralized feedback mechanism may be due to spatial decentralization in the large structure, or it may be due to an imposed structure dictated by the designer, so that the channel-by-channel feedback is simpler to implement and faster to compute in real time.

The well known transformation $q = \Phi \zeta$, where Φ is a matrix of modal vectors (the mode shape matrix), is now utilized as discussed in Section 2.2.1 to simultaneously diagonalize M and K in (2.51) to give the state space formulation

$$\frac{d}{dt} \begin{bmatrix} \zeta \\ \dot{\zeta} \end{bmatrix} = \begin{bmatrix} 0 & I \\ -\Omega^2 & 0 \end{bmatrix} \begin{bmatrix} \zeta \\ \dot{\zeta} \end{bmatrix} + \sum_{i=1}^N \begin{bmatrix} 0 \\ \Phi^T M^{-1} B_i \end{bmatrix} u_i \quad (2.52a)$$

$$y_i = [C_{p_i} \Phi \quad C_{v_i} \Phi] \begin{bmatrix} \zeta \\ \dot{\zeta} \end{bmatrix} \quad (2.52b)$$

We note that in the formulation (2.52a) no damping is present in the structure; for our purposes, a term to account for damping effects may be easily included in place of the lower-right zero of the first matrix in (2.52a). Typically, damping ratios on the order of 0.5% are assumed as the damping effects are very small for large flexible structures. Furthermore, actuator and sensor dynamics can also be appended to this formulation.

2.3.3 Centralized and Decentralized Optimal Output Feedback

Consider the basic *decentralized quadratic regulator problem*:

$$\dot{x} = Ax + \sum_{i=1}^{\nu} B_i u_i \quad x(0) = x_0 \quad (2.53)$$

$$y_i = C_i x + D_i u_i \quad i = 1, \dots, \nu \quad (2.54)$$

with cost

$$J = \int_0^{\infty} (x^T Q x + 2 \sum_{i=1}^{\nu} x^T N_i u_i + \sum_{i=1}^{\nu} u_i^T R_i u_i) dt \quad (2.55)$$

and the following feedback structure constraint:

$$u_i = K_i y_i \quad i = 1, \dots, \nu. \quad (2.56)$$

Remark 1: In the following analysis, the terms corresponding to a direct feedthrough of the inputs to the outputs (i.e., D_i) are neglected for notational convenience. If direct feedthroughs are present, outputs of the forms $y_i^{new} = y_i - D_i u_i$ can be utilized.

It can be shown that the necessary conditions for minimizing J given by (2.55) with the controller structure (2.56) imply the solution of the following system of nonlinear algebraic equations:

$$\begin{cases} A_c^T P + P A_c + \bar{Q} = 0 \\ A_c L + L A_c^T + X_0 = 0 \end{cases} \quad (2.57)$$

and

$$\nabla_{K_i} J = B_i^T P L C_i^T + R_i K_i C_i L C_i^T + N_i^T L C_i^T = 0 \quad i = 1, \dots, \nu \quad (2.58)$$

where

$$A_c = A + \sum_{i=1}^{\nu} B_i K_i C_i \quad (2.59)$$

$$\bar{Q} = Q + \sum_{i=1}^{\nu} C_i^T K_i^T R_i K_i C_i + \sum_{i=1}^{\nu} (N_i K_i C_i + C_i^T K_i^T N_i^T) \quad (2.60)$$

$$X_0 = x_0 x_0^T. \quad (2.61)$$

A software package has been developed recently [58] to solve this problem and has been used in all the numerical examples that follow. Many approaches have been

introduced in the literature to solve the above system of equations. The approach pursued in our software package is a gradient method which gives at iteration j

$$K_i^{j+1} = K_i^j - \epsilon \nabla_{K_i^j} J \quad i = 1, \dots, \nu \quad (2.62)$$

where ϵ is a step size chosen by appealing to the Fibonacci linear search technique.

The solution methodology is as follows:

- i) Choose an initial stabilizing gain and a tolerance δ on the norm of the gradient $\nabla_{K_i} J$.
- ii) Solve the Lyapunov equations given by (2.57).
- iii) Find the new gain using (2.62) and the Fibonacci linear search.
- iv) Go back to step (ii) if $\|\nabla_{K_i^j} J\| \geq \delta$.
- v) Stop.

Extensions of the basic problem can be made by considering the following decentralized dynamic stabilizing compensator:

$$\dot{z}_i = F_i z_i + G_i y_i$$

$$u_i = H_i z_i + M_i y_i \quad i = 1, \dots, \nu.$$

The case of dynamic output feedback is reduced to the previous problem by augmenting the state of the compensators with that of the system, i.e.,

$$\begin{bmatrix} \dot{x} \\ z_1 \\ z_2 \\ \vdots \\ z_\nu \end{bmatrix} = \left[\begin{array}{c|c} A & 0 \\ \hline 0 & 0 \end{array} \right] \begin{bmatrix} x \\ z_1 \\ z_2 \\ \vdots \\ z_\nu \end{bmatrix} + \sum_{i=1}^{\nu} \left[\begin{array}{c|c} B_i & 0 \\ \hline 0 & 0 \\ 0 & I \\ 0 & 0 \end{array} \right] \begin{bmatrix} u_i \\ \bar{v}_i \end{bmatrix},$$

$$\bar{y}_i = \left[\begin{array}{c|c|c|c} C_i & 0 & 0 & 0 \\ \hline 0 & 0 & \bar{C}_i & 0 \end{array} \right] \begin{bmatrix} x \\ z_1 \\ z_2 \\ \vdots \\ z_\nu \end{bmatrix}$$

where $\bar{v}_i = \dot{z}_i$ and \hat{C}_i is provided by the designer; modifying the cost to

$$J = \frac{1}{2} \int_0^\infty (x^T Q x + 2 \sum_{i=1}^{\nu} x^T N_i u_i + \sum_{i=1}^{\nu} u_i^T R_i^1 u_i + \sum_{i=1}^{\nu} \dot{z}_i^T R_i^2 \dot{z}_i) dt.$$

Further extensions of the above problem can be made by incorporating frequency weighting into the quadratic criterion. Centralized frequency weighting has been introduced in [5]. A decentralized frequency weighted cost was mentioned in [56] and has recently been further developed [59]. We now outline some of the steps necessary in reducing the decentralized frequency weighting to the decentralized LQR problem.

2.3.4 Decentralized Frequency Shaping (DFS)

Frequency shaping permits frequency dependent weighting matrices in the quadratic cost functional [5]. This allows the designer to use some of the classical frequency domain constraints in the state-space LQ setting. The cost functional in the frequency domain becomes:

$$J = \frac{1}{2} \int_{-\infty}^{\infty} \left(X^* Q_0 X + \sum_{i=1}^{\nu} [Y_i^*(j\omega) Q_i Y_i(j\omega) + U_i^*(j\omega) R_i(\omega) U_i(\omega)] \right) d\omega \quad (2.63)$$

where $Q_0 \geq 0$ and Q_i and R_i for $i = 1, \dots, \nu$ are positive semidefinite and positive definite Hermitian matrices respectively.

Frequency shaping is used in the following way. As presented in [60], frequency dependent weighting on the control allows for better robustness properties than similar weighting on the state, so the control input is chosen for the frequency dependent cost weighting and the output weighting remains constant. By choosing $R_i(j\omega)$ to be larger for large ω than for small ω , control actions are penalized at high frequencies more than low frequencies. Effectively, this choice of $R_i(j\omega)$ reduces the loop gain of the i^{th} channel at high frequencies which, in turn, results in a system that can better accommodate high frequency uncertainties related to channel i .

Using the notation of [59], Q_i and R_i are factored as:

$$Q_i = \hat{Q}_i^* \hat{Q}_i \quad (2.64)$$

$$R_i = \hat{R}_i^* \hat{R}_i \quad i = 1, \dots, \nu. \quad (2.65)$$

For this decomposition, the cost may be written as

$$J = \int_{-\infty}^{\infty} \left(X^* Q_0 X + \sum_{i=1}^{\nu} [\hat{Y}_i^* \hat{Y}_i + \hat{U}_i^* \hat{U}_i] \right) d\omega \quad (2.66)$$

where

$$\begin{aligned}\hat{Y}_i &= \hat{Q}_i Y_i, \\ \hat{U}_i &= \hat{R}_i U_i\end{aligned}\quad i = 1, \dots, \nu.$$
(2.67)

Assuming the \hat{Q}_i are proper rational matrices and the \hat{R}_i are rational matrices with entries being either proper or non-causal, they can be realized in the following manner respectively:

$$\begin{aligned}\dot{\eta}_i &= F_i^y \eta_i + G_i^y y_i \\ \hat{Y}_i &= H_i^y \eta_i + N_i^y y_i\end{aligned}\quad i = 1, \dots, \nu$$
(2.68)

and

$$\begin{aligned}\dot{z}_i &= F_i^u z_i + G_i^u u_i \\ \hat{U}_i &= H_i^u z_i + \sum_{p=1}^k N_i^p u_i^{(p)}\end{aligned}\quad i = 1, \dots, \nu.$$
(2.69)

where $(\cdot)^{(p)}$ denotes the p th derivative of (\cdot) with respect to time. Furthermore, it is assumed that the entries of $R_i(\omega)$ have k more zeros than poles. For additional results in decentralized frequency shaping, see [59].

2.3.5 Overlapping Decompositions

Due to the complexity and large scale of systems describing flexible structures, decomposition methods play a significant role for problem solving. One method for controller design of interconnected systems is a decentralized approach which assumes that each subsystem is controlled independently using local information and performance considerations. This method deals with *disjoint* subsystems, where there is no information overlap in either the dynamics, control, or output. However, for a large class of systems (e.g., flexible structures, traffic regulation, and power systems), the sharing of information among controllers is essential. For systems of this type, it is natural to consider an overlapping decomposition approach, where subsystems can have overlapping of states, inputs, and outputs.

The underlying concept for overlapping decomposition is the inclusion principle. In the next section, we introduce the inclusion principle and several concepts relating to inclusion such as expansion/contraction and restriction/unrestriction. Using these

concepts, controllers are designed for the individual (disjoint) subsystems, then contracted to the original state-space. Conditions for contraction of controllers are also discussed.

The Inclusion Principle

The mathematical framework for decentralized control using overlapping decomposition is the *Inclusion Principle*. Here, overlapping subsystems are expanded into a larger state space and control space, such that the subsystems appear to be decoupled in state, input, and output. Control laws are then designed for these smaller, disjoint subsystems. Using certain *contraction conditions*, the subsystems and controllers are contracted to the original space for implementation, and if all of the appropriate conditions are met, then the closed loop properties that were derived for the expanded decoupled subsystems are intact when the system is contracted. For this structure, we consider the state, input, and output inclusion problem for controller design.

Consider a system with two decentralized control agents described by

$$\begin{aligned} \dot{x} &= Ax + B_1 u_1 + B_2 u_2, \quad x(0) = x_0 \in \mathbb{R}^n \\ y_1 &= C_1 x \\ y_2 &= C_2 x \end{aligned} \tag{2.70}$$

where $x \in \mathbb{R}^n$ is the state, $u_i \in \mathbb{R}^{m_i}$ is the input, and $y_i \in \mathbb{R}^{p_i}$ is the output of the i th control agent ($i = 1, 2$). It is assumed that the system given above does not have any decentralized fixed modes (DFM's) [61].

Consider the following partition of the state x :

$$x = \begin{bmatrix} x_{o_1 c_1} \\ x_{oc} \\ x_{o_2 c_2} \end{bmatrix} \tag{2.71}$$

where $x_{o_i c_i} \in \mathbb{R}^{n_{o_i c_i}}$ is observable and controllable only by agent- i , and $x_{oc} \in \mathbb{R}^{n_{oc}}$ is observable and controllable by both agents. Then the matrices of (2.70) have the following forms [62]:

$$A = \begin{bmatrix} A_{11} & 0 & 0 \\ A_{21} & A_{22} & A_{23} \\ 0 & 0 & A_{33} \end{bmatrix}$$

$$B \triangleq [B_1 : B_2] = \begin{bmatrix} B_{11} & : & 0 \\ B_{21} & : & B_{22} \\ 0 & : & B_{32} \end{bmatrix} \quad (2.72)$$

$$C \triangleq \begin{bmatrix} C_1 \\ \dots \\ C_2 \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & 0 \\ \dots & \dots & \dots \\ 0 & C_{22} & C_{23} \end{bmatrix}$$

Consider the transformation:

$$\hat{x} = Vx = \begin{bmatrix} x_{o_1 c_1} \\ x_{oc} \\ \dots \\ x_{oc} \\ x_{oc} \end{bmatrix} \in \mathbb{R}^{\hat{n}}, \quad \hat{n} = n + n_{oc} \quad (2.73)$$

The expansion [7] of the original system (2.70) with respect to the transformation (2.73) is given by (see [9, 14] for developments):

$$\begin{aligned} \dot{\hat{x}} &= \hat{A}\hat{x} + \hat{B}u, \\ y &= \hat{C}\hat{x} \end{aligned} \quad (2.74)$$

where

$$\hat{x}(0) = Vx_0 \in \mathbb{R}^{\hat{n}} \quad (2.75)$$

and

$$\hat{A} = \begin{bmatrix} A_{11} & 0 & \vdots & 0 & 0 \\ A_{12} & A_{22} & \vdots & 0 & A_{13} \\ \dots & \dots & \dots & \dots & \dots \\ A_{12} & 0 & \vdots & A_{22} & A_{13} \\ 0 & 0 & \vdots & 0 & A_{33} \end{bmatrix} \triangleq \begin{bmatrix} \hat{A}_1 & \hat{A}_{12} \\ \hat{A}_{21} & \hat{A}_2 \end{bmatrix}$$

$$\hat{B} = \begin{bmatrix} B_{11} & \vdots & 0 \\ B_{21} & \vdots & B_{22} \\ \dots & \dots & \dots \\ B_{21} & \vdots & B_{22} \\ 0 & \vdots & B_{32} \end{bmatrix} \triangleq \begin{bmatrix} \hat{B}_1 & \hat{B}_{12} \\ \hat{B}_{21} & \hat{B}_2 \end{bmatrix} \quad (2.76)$$

$$\hat{C} = \begin{bmatrix} C_{11} & C_{12} & \vdots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \vdots & C_{22} & C_{23} \end{bmatrix} \triangleq \begin{bmatrix} \hat{C}_1 & \hat{C}_{12} \\ \hat{C}_{21} & \hat{C}_2 \end{bmatrix}$$

It can be shown [7] that the response $x(t)$ of (2.70) is related to the response $\hat{x}(t)$ of (2.74) by

$$x(t) = V^I \hat{x}(t) \quad \forall t \geq 0 \quad (2.77)$$

where V^I is the generalized inverse of V satisfying $V^I V = I$. Furthermore, (2.70) and (2.74) have equivalent input/output descriptions, i.e.,

$$\hat{G}(s) \triangleq \hat{C}(sI - \hat{A})^{-1} \hat{B} \equiv G(s) \triangleq C(sI - A)^{-1} B \quad (2.78)$$

The systems are related by the following transformations from [9]:

$$\begin{aligned} \hat{A} &= V A U + M \\ \hat{B} &= V B Q + N \\ \hat{C} &= T C U + L \end{aligned} \quad (2.79)$$

where M, N and L are complimentary matrices of appropriate dimensions. For (2.74) to be an expansion of (2.70), a proper choice of M, N and L is required as given in [9]. In this case, the transformation matrices are given as follows:

$$\begin{aligned} V &= \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}, \\ U &= V^I, \\ M &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \frac{A_{22}}{2} & -\frac{A_{22}}{2} & 0 \\ 0 & -\frac{A_{22}}{2} & \frac{A_{22}}{2} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \\ Q &= I, \\ N &= 0, \\ T &= I, \\ L &= \begin{bmatrix} 0 & \frac{C_{12}}{2} & -\frac{C_{12}}{2} & 0 \\ 0 & -\frac{C_{22}}{2} & \frac{C_{22}}{2} & 0 \end{bmatrix}, \end{aligned}$$

Now consider the *design model* described by:

$$\begin{aligned}\dot{\tilde{x}} &= \tilde{A}\tilde{x} + \tilde{B}u \\ y &= \tilde{C}\tilde{x}\end{aligned}\tag{2.80}$$

where

$$\begin{aligned}\tilde{A} &= \begin{bmatrix} \hat{A}_1 & 0 \\ 0 & \hat{A}_2 \end{bmatrix}, \\ \tilde{B} &= \begin{bmatrix} \hat{B}_1 & 0 \\ 0 & \hat{B}_2 \end{bmatrix}, \\ \tilde{C} &= \begin{bmatrix} \hat{C}_1 & 0 \\ 0 & \hat{C}_2 \end{bmatrix}\end{aligned}\tag{2.81}$$

The transfer function matrix (TFM) for the design model (2.80) is given by:

$$\tilde{G}(s) \triangleq \tilde{C}(sI - \tilde{A})^{-1}\tilde{B} = \text{block diag } [G_1(s), G_2(s)]\tag{2.82}$$

where $G_i(s)$ is the TFM for the decoupled subsystem:

$$\begin{aligned}\dot{x}_i &= \hat{A}_i x_i + \hat{B}_i u_i \\ y_i &= \hat{C}_i x_i, \quad i = 1, 2\end{aligned}\tag{2.83}$$

Note that (2.83) is the minimal local model for agent- i .

Model Reduction

The decision of which modes to retain from each subsystem can be made by considering the cross grammians as discussed in Section 2.2.2. A decentralized control strategy for a LSS is arrived at by the following two tasks: (1) decide on an appropriate system decomposition, and (2) truncate the model to a manageable number of modes. Both of these goals can be realized by using a *balanced realization* [46]. In the decentralized case, we can perform a decentralized balanced realization for each I/O channel, and perform modal truncation on each subsystem by deleting those modes that are weakly controllable (observable) through that particular I/O channel. Thus, a decentralized balanced realization not only gives the control designer insight for model reduction, it also helps to decide on the appropriate subsystem decomposition.

In [63], Williams presents an efficient algorithm for computation of grammians for models given in modal coordinates. In his work, decentralized control is assumed, where control is affected through a given I/O pair (u_i, y_i) , or $u_i = G_i(s)y_i(s)$.

2.3.6 Isolated Boundary Loading and Controlled Component Synthesis (CCS)

Isolated Boundary Loading

In Component Mode Synthesis (CMS) [64, 65, 66], a large flexible structure is assumed to be composed of components or substructures. Each component or substructure is separately modeled using finite element modeling techniques. The order of each of these component models is reduced using the assumed-modes method or the Rayleigh-Ritz method with constraint equations. The reduced order components are then mathematically coupled together with interface compatibility equations to form a reduced order model of the entire structure.

Component modes can be classified as rigid-body modes, normal modes, constraint modes, or attachment modes [67, 68]. The basic idea in CMS is to find a set of component modes for each component that make the greatest contribution to the modes of the composite structure. Several methods exist to select these modes. Perhaps the most widely known methods are the Craig-Bampton method [69], the MacNeal-Rubin method [70, 71], and the Benfield-Hruda method [72].

Isolated boundary loading developed by Young [13] is a component modeling method which is based on the boundary loading ideas of the Benfield-Hruda method. Dynamic contributions of other components are included in the model of a given component.

This discussion will be restricted to a system comprised of only two components. Superscripts A and B will be used when referring to quantities associated with the two components. Each of these components is assumed to have dimension n . In general, any number of components containing any number of nodes may be used.

CMS methods separate the component nodes into two groups, interface nodes and internal nodes. Here, the internal nodes are further separated into internal boundary nodes and internal nodes. The internal boundary nodes are the nodes of the boundary element that are not actually connected to the adjoining structure; i.e., the nodes that connect the boundary elements to the internal elements of the same component. The remaining nodes are the internal nodes. Thus, the finite element model of a component can be partitioned as

$$\begin{bmatrix} M_{ii} & M_{ik} & 0 \\ M_{ki} & M_{kk} & M_{kj} \\ 0 & M_{jk} & M_{jj} \end{bmatrix} \begin{bmatrix} \tilde{q}_i \\ \tilde{q}_k \\ \tilde{q}_j \end{bmatrix} + \begin{bmatrix} K_{ii} & K_{ik} & 0 \\ K_{ki} & K_{kk} & K_{kj} \\ 0 & K_{jk} & K_{jj} \end{bmatrix} \begin{bmatrix} q_i \\ q_k \\ q_j \end{bmatrix} = \begin{bmatrix} f_i \\ f_k \\ f_j \end{bmatrix} \quad (2.54)$$

where q_i , q_k , and q_j are the generalized displacements at the interface, internal boundary, and internal nodes, respectively.

In isolated boundary loading, a truncation matrix is used to eliminate the internal and internal boundary nodes. The truncation matrix is given by

$$\Psi_{ni}^B = \begin{bmatrix} I_{ii} \\ 0 \\ 0 \end{bmatrix}. \quad (2.85)$$

Thus, the interface loading matrices are merely the stiffness and mass matrices corresponding to the boundary nodes of the adjoining component:

$$K_{ii}^B = \Psi_{ni}^{B^T} K_{nn}^B \Psi_{ni}^B \quad (2.86)$$

$$M_{ii}^B = \Psi_{ni}^{B^T} M_{nn}^B \Psi_{ni}^B. \quad (2.87)$$

The resulting modified stiffness and mass matrices for component *A* are given by

$$K_{nn}^A = \begin{bmatrix} K_{ii}^A & K_{ik}^A & 0 \\ K_{ki}^A & K_{kk}^A & K_{kj}^A \\ 0 & K_{jk}^A & K_{jj}^A \end{bmatrix} + \begin{bmatrix} \Psi_{ni}^{B^T} K_{nn}^B \Psi_{ni}^B & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (2.88)$$

$$M_{nn}^A = \begin{bmatrix} M_{ii}^A & M_{ik}^A & 0 \\ M_{ki}^A & M_{kk}^A & M_{kj}^A \\ 0 & M_{jk}^A & M_{jj}^A \end{bmatrix} + \begin{bmatrix} \Psi_{ni}^{B^T} M_{nn}^B \Psi_{ni}^B & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (2.89)$$

which reduce to

$$K_{nn}^A = \begin{bmatrix} K_{ii}^A + K_{ii}^B & K_{ik}^A & 0 \\ K_{ki}^A & K_{kk}^A & K_{kj}^A \\ 0 & K_{jk}^A & K_{jj}^A \end{bmatrix} \quad (2.90)$$

$$M_{nn}^A = \begin{bmatrix} M_{ii}^A + M_{ii}^B & M_{ik}^A & 0 \\ M_{ki}^A & M_{kk}^A & M_{kj}^A \\ 0 & M_{jk}^A & M_{jj}^A \end{bmatrix}. \quad (2.91)$$

By letting the modified boundary submatrices be

$$K_{iiL}^A = K_{ii}^A + K_{ii}^B \quad (2.92)$$

$$M_{iiL}^A = M_{ii}^A + M_{ii}^B, \quad (2.93)$$

Equations (2.90) and (2.91) can finally be written as

$$K_{nn}^A = \begin{bmatrix} K_{iiL}^A & K_{ik}^A & 0 \\ K_{ki}^A & K_{kk}^A & K_{kj}^A \\ 0 & K_{jk}^A & K_{jj}^A \end{bmatrix} \quad (2.94)$$

$$\mathcal{M}_{nn}^A = \begin{bmatrix} M_{ii/L}^A & M_{ik}^A & 0 \\ M_{ki}^A & M_{kk}^A & M_{kj}^A \\ 0 & M_{jk}^A & M_{jj}^A \end{bmatrix}. \quad (2.95)$$

Similarly, the modified stiffness and mass matrices for component B can be written as

$$\mathcal{K}_{nn}^B = \begin{bmatrix} K_{ii/L}^B & K_{ik}^B & 0 \\ K_{ki}^B & K_{kk}^B & K_{kj}^B \\ 0 & K_{jk}^B & K_{jj}^B \end{bmatrix} \quad (2.96)$$

$$\mathcal{M}_{nn}^B = \begin{bmatrix} M_{ii/L}^B & M_{ik}^B & 0 \\ M_{ki}^B & M_{kk}^B & M_{kj}^B \\ 0 & M_{jk}^B & M_{jj}^B \end{bmatrix} \quad (2.97)$$

where

$$K_{ii/L}^B = K_{ii/L}^A \quad (2.98)$$

$$M_{ii/L}^B = M_{ii/L}^A. \quad (2.99)$$

Equations (2.99) and (2.98) indicate that the modified boundary submatrices are the same for each of the components involved and may be found either by loading component A onto component B or by loading component B onto component A .

It should be noted that the modified mass and stiffness matrices can be determined from the model of an expanded component consisting of the original component extended one element into the adjoining component [13]. The model of the expanded component will contain the internal boundary nodes of the adjoining component in addition to all of the nodes of the original component. The modified mass and stiffness matrices are then found by deleting the rows and columns associated with the internal boundary nodes of the adjoining component.

The coupling of substructures using isolated boundary loading employs superposition to couple the components at the boundary. Essentially, the equation of motion for the boundary of the coupled structure is found from superposition of the forces and motions at the boundaries of the individual components A and B such that

$$q_i^{AB} = q_i^A + q_i^B \quad (2.100)$$

$$f_i^{AB} = f_i^A + f_i^B. \quad (2.101)$$

The finite element model for the coupled structure is given by

$$\mathcal{M}^{AB} \ddot{q}^{AB} + \mathcal{K}^{AB} q^{AB} = f^{AB} \quad (2.102)$$

where

$$K^{AB} = \begin{bmatrix} K_{jj}^A & K_{jk}^A & 0 & 0 & 0 \\ K_{kj}^A & K_{kk}^A & K_{ki}^A & 0 & 0 \\ 0 & K_{ik}^A & K_{ii}^A + K_{ii}^B & K_{ik}^B & 0 \\ 0 & 0 & K_{ki}^B & K_{kk}^B & K_{kj}^B \\ 0 & 0 & 0 & K_{jk}^B & K_{jj}^B \end{bmatrix} \quad (2.103)$$

$$M^{AB} = \begin{bmatrix} M_{jj}^A & M_{jk}^A & 0 & 0 & 0 \\ M_{kj}^A & M_{kk}^A & M_{ki}^A & 0 & 0 \\ 0 & M_{ik}^A & M_{ii}^A + M_{ii}^B & M_{ik}^B & 0 \\ 0 & 0 & M_{ki}^B & M_{kk}^B & M_{kj}^B \\ 0 & 0 & 0 & M_{jk}^B & M_{jj}^B \end{bmatrix} \quad (2.104)$$

$$q^{AB} = [q_j^A \quad q_k^A \quad q_i^{AB} \quad q_k^B \quad q_j^B]^T \quad (2.105)$$

$$f^{AB} = [f_j^A \quad f_k^A \quad f_i^{AB} \quad f_k^B \quad f_j^B]^T. \quad (2.106)$$

Using isolated boundary loading for component modeling and this superposition-based approach for substructure coupling, the coupled model is the *exact* model for the composite system. In conventional large-scale system design methods, the interactions between components are generally discarded in decoupled substructure models. The component models formed using isolated boundary loading, however, include the interactions with other components. The component models are identical to the decoupled subsystem models obtained using overlapping decomposition [7, 8, 9, 10] on the composite system model. Thus, the powerful tools used in overlapping decomposition analysis of large-scale systems may be applied to the isolated boundary loading models.

Controlled Component Synthesis

Controlled component synthesis (CCS) is a control design method developed by Young [13] in which controllers are designed for individual components based on component models as discussed previously. Using conventional large-scale system design methods requires a full model of the entire system. This model is typically decomposed into interconnected subsystem models which are then decoupled by ignoring subsystem interaction terms. A key advantage of controlled component synthesis is that a full model of the entire structure is not required. This decentralized control strategy extends the component mode synthesis modeling concepts into control concepts by allowing component controllers based on component models to be developed independently.

Overlapping decomposition can be used to derive component models from the complete model of a structure. Considering the two-component system example from the previous section, the component model is given by

$$\begin{bmatrix} M_{ii}^A & M_{ik}^A & 0 \\ M_{ki}^A & M_{kk}^A & M_{kj}^A \\ 0 & M_{jk}^A & M_{jj}^A \end{bmatrix} \begin{bmatrix} \tilde{q}_i^A \\ \tilde{q}_k^A \\ \tilde{q}_j^A \end{bmatrix} + \begin{bmatrix} K_{ii}^A & K_{ik}^A & 0 \\ K_{ki}^A & K_{kk}^A & K_{kj}^A \\ 0 & K_{jk}^A & K_{jj}^A \end{bmatrix} \begin{bmatrix} q_i^A \\ q_k^A \\ q_j^A \end{bmatrix} = \begin{bmatrix} f_i^A \\ f_k^A \\ f_j^A \end{bmatrix} + \begin{bmatrix} M_{ik}^B \tilde{q}_k^B + K_{ik}^B q_k^B + f_i^B \\ 0 \\ 0 \end{bmatrix}. \quad (2.107)$$

The second term on the right side of this equation represents the dynamic contributions of the internal boundary motion of component *B* to the motion of component *A*. Neglecting this term results in the component model for *A* which would be derived using isolated boundary loading.

The interlocking control concept is one of the key ideas behind CCS. With interlocking control, colocated sensors are placed at the internal boundary coordinates of each component. The component controllers are designed using the component models as derived in the previous section to minimize motion at the internal boundary nodes. Essentially, one or more controllers are inserted near the substructure interfaces as shown in Figure 2.1. By minimizing the internal boundary motion, the

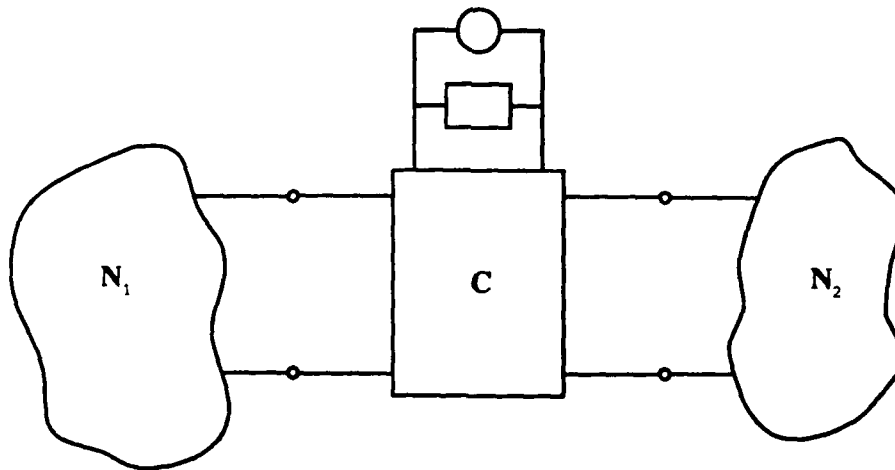


Figure 2.1: Inserting a Controller

transmission of disturbances between components is suppressed, the second term in Equation (2.107) becomes small, and the accuracy of the assumed component model increases.

Although any technique may be used for controller design, the linear quadratic optimal regulator (LQR) is perhaps the most convenient [13]. The weighted internal boundary states are to be included in the cost function with the weighted control inputs. Because the sensors are located at the internal boundary coordinates, these states are merely the outputs of the system. The control inputs are $u = f_k$ because the actuators are also located at the internal boundary coordinates. Using the LQR approach, the component model for component A is formed as

$$\begin{bmatrix} M_{iLL}^A & M_{ik}^A & 0 \\ M_{ki}^A & M_{kk}^A & M_{kj}^A \\ 0 & M_{jk}^A & M_{jj}^A \end{bmatrix} \begin{bmatrix} \ddot{q}_i^A \\ \ddot{q}_k^A \\ \ddot{q}_j^A \end{bmatrix} + \begin{bmatrix} K_{iLL}^A & K_{ik}^A & 0 \\ K_{ki}^A & K_{kk}^A & K_{kj}^A \\ 0 & K_{jk}^A & K_{jj}^A \end{bmatrix} \begin{bmatrix} q_i^A \\ q_k^A \\ q_j^A \end{bmatrix} = \begin{bmatrix} 0 \\ u^A \\ 0 \end{bmatrix} \quad (2.108)$$

where u^A is the control input. Using velocity measurements, the output is defined to be

$$y^A = \dot{q}_k^A. \quad (2.109)$$

The control law for component A is found to minimize the cost function

$$J = \frac{1}{2} \int_0^\infty (y^{AT} Q^A y^A + u^{AT} R^A u^A) dt. \quad (2.110)$$

In general, a state feedback control law could be found to minimize this performance criterion. However, optimal output feedback can also be employed to achieve good results and is easier to implement than state feedback. This procedure is repeated for each component. CCS controller designs are inherently decentralized since only the component outputs or states are required for the component control law.

2.3.7 Maximum Entropy/Optimal Projection (MEOP)

One of the fundamental concerns in the application of modern control designs for active vibration suppression is the loss of stability and the performance degradation in face of modeling errors. This is especially true in the area of flexible structures which are inherently infinite-dimensional as discussed in Section 2.1.1. The truncation of the infinite dimensional system to a finite dimensional system (as in the finite element representation of Section 2.1.2) induces truncation errors which tend to be more significant at high frequencies. Therefore, the theoretical (calculated) mode shapes and frequencies are less accurate for higher-order modes. This can be interpreted as the higher modal frequencies and mode shapes are more sensitive to small details of geometry, construction, and material properties. The high frequency modes are thus subjected to larger errors in a finite element model.

To tackle this kind of control problem, Hyland [73, 74, 75] proposed the stochastic Maximum Entropy (ME) approach to handle model uncertainties in order to improve the system robustness with respect to parameter variations and high order unmodelled dynamics. The ME approach to model uncertainty is essentially induced by the minimum data set which incorporates as little statistical parameter information as possible consistent with the available *a priori* statistical data. The basic premise of the Maximum Entropy error modeling is that the magnitude of the error is a white noise process. The justification for the use of the Maximum Entropy stochastic approach for structural control design is reinforced by various qualitative results for the mean square optimization problem examined in [75].

From the view point of control engineers, the ME approach was developed as a means to design compensators with the performance/robustness trade-off being explicitly determined by the quadratic cost functional. Thus, the ME algorithm is essentially a "robust" extension to LQG theory. By explicitly including the parameter uncertainties in the model, the performance/robustness trade-off can be directly manipulated to suit the design requirements.

In addition to the potential problem of modeling errors in flexible structures, the large dimensionality of the dynamic model inherent in the structural system poses another problem in control design and implementation. It is known that the classical LQG controllers have the same order as the controlled plant. If the plant is of high order, then the LQG controller dimension may be too large for real time implementation. Reduced order controller design has therefore received much attention in the past few years for flexible structure applications. In the literature, various design algorithms appear which involve either ad hoc reduction and truncation schemes, or the two-step approach to reduced order controller design in which a compensator order reduction is preceded by the LQG design or a LQG design is preceded by the plant order reduction [46, 76, 77, 78, 79, 80]. However, these approaches are not optimal, and worse yet, they may cause closed loop instability [81]. The Optimal Projection (OP) approach proposed by Hyland and Bernstein [82] focuses on developing direct reduced order design methods which have greater flexibility with respect to the dimension of the controller. It involves the minimization of a quadratic cost functional while directly constraining the order of the feedback controller to guarantee the optimality of the reduced order design.

In order to design a controller which takes care of both the robustness issue and the controller dimensionality issue, the combined Maximum Entropy/Optimal Projection (MEOP) methodology has been developed[83]. This methodology provides a powerful, yet very structured algorithm for designing a robust, fixed-order dynamic compensator.

Consider an n^{th} order linear time invariant system

$$\dot{x} = Ax + Bu + w_1 \quad y = Cx + w_2 \quad , \quad (2.111)$$

under the usual assumptions of controllability and observability, where w_1 , w_2 are zero mean white noise processes with noise intensity matrices V_1 and V_2 , respectively. It is required to design an n_c^{th} order *robust*, zero set-point compensator

$$\dot{x}_c = A_c x_c + Fy \quad u = -Kx_c \quad (2.112)$$

to minimize the cost functional

$$J = \lim_{\tau \rightarrow \infty} \frac{1}{\tau} E \left\{ \int_0^\tau (x^T(t)R_1x(t) + u^T(t)R_2u(t))dt \right\} \quad (2.113)$$

for $x \in \mathbb{R}^n$, $x_c \in \mathbb{R}^{n_c}$, $u \in \mathbb{R}^m$, $y \in \mathbb{R}^l$, $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{l \times n}$, $A_c \in \mathbb{R}^{n_c \times n_c}$, $F \in \mathbb{R}^{n_c \times l}$, $K \in \mathbb{R}^{m \times n}$, $R_1 \in \mathbb{R}^{n \times n}$ (positive semi-definite), and $R_2 \in \mathbb{R}^{m \times m}$ (positive definite).

The first order necessary conditions for the quadratically optimal, steady state, robust, reduced-order dynamic compensation are the existence of non-negative definite matrices $P \in \mathbb{R}^{n \times n}$, $Q \in \mathbb{R}^{n \times n}$, $\hat{P} \in \mathbb{R}^{n \times n}$, $\hat{Q} \in \mathbb{R}^{n \times n}$ satisfying the following coupled Lyapunov and Riccati equations:

$$\begin{aligned} 0 = & PA_s + A_s^T P + \sum_{i=1}^{\mu} A_i^T P A_i - P_s^T R_{2s}^{-1} P_s + R_1 \\ & + \sum_{i=1}^{\mu} (A_i - Q_s V_{2s}^{-1} C_i)^T \hat{P} (A_i - Q_s V_{2s}^{-1} C_i) \\ & + \tau_{\perp}^T P B R_2^{-1} B^T P \tau_{\perp} \end{aligned} \quad (2.114)$$

$$\begin{aligned} 0 = & A_s Q + Q A_s^T + \sum_{i=1}^{\mu} A_i Q A_i^T - Q_s V_{2s}^{-1} Q_s^T + V_1 \\ & + \sum_{i=1}^{\mu} (A_i - B_i R_{2s}^{-1} P_s) \hat{Q} (A_i - B_i R_{2s}^{-1} P_s)^T \\ & + \tau_{\perp} Q C^T V_2^{-1} C Q \tau_{\perp}^T \end{aligned} \quad (2.115)$$

$$0 = \hat{P} A_{QS} + A_{QS}^T \hat{P} + P_s^T R_{2s}^{-1} P_s - \tau_{\perp}^T P B R_2^{-1} B^T P \tau_{\perp} \quad (2.116)$$

$$0 = A_{PS} \hat{Q} + \hat{Q} A_{PS}^T + Q_s V_{2s}^{-1} Q_s^T - \tau_{\perp} Q C^T V_2^{-1} C Q \tau_{\perp}^T \quad (2.117)$$

where the projection operator $\tau \in \mathbb{R}^{n \times n}$ is given by

$$\tau = \sum_{k=1}^{n_c} \prod_k [\hat{Q} \hat{P}]$$

and $\Pi_k[\hat{Q}\hat{P}]$ represents the k^{th} eigenprojection of $\hat{Q}\hat{P}$, with $\tau_\perp = I_n - \tau$. In (2.114)–(2.117), $A_i \in \mathbb{R}^{n \times n}$, $B_i \in \mathbb{R}^{n \times m}$, $C_i \in \mathbb{R}^{l \times n}$ are the uncertainty matrices of the plant [83], from which

$$\begin{aligned} A_s &\equiv A + \frac{1}{2} \sum_{i=1}^{\mu} A_i^2 & B_s &\equiv B + \frac{1}{2} \sum_{i=1}^{\mu} A_i B_i \\ C_s &\equiv C + \frac{1}{2} \sum_{i=1}^{\mu} C_i A_i & R_{2s} &\equiv R_2 + \sum_{i=1}^{\mu} B_i^T (P + \hat{P}) B_i \\ V_{2s} &\equiv V_2 + \sum_{i=1}^{\mu} C_i (Q + \hat{Q}) C_i^T & A_{QS} &\equiv A_s - Q_s V_{2s}^{-1} C_s \\ P_s &\equiv B_s^T P + \sum_{i=1}^{\mu} B_i^T (P + \hat{P}) A_i & A_{PS} &\equiv A_s - B_s R_{2s}^{-1} P_s \\ Q_s &\equiv Q C_s^T + \sum_{i=1}^{\mu} A_i (Q + \hat{Q}) C_i^T \end{aligned}$$

where μ is the number of sets of uncorrelated uncertainties.

In terms of the solution to the above MEOP equations, the compensator dynamics are specified according to

$$A_c = \Gamma(A_s - Q_s V_{2s}^{-1} C_s - B_s R_{2s}^{-1} P_s) G^T \quad (2.118)$$

$$F = \Gamma Q_s V_{2s}^{-1} \quad (2.119)$$

$$K = R_{2s}^{-1} P_s G^T \quad (2.120)$$

where the operators $G \in \mathbb{R}^{n_c \times n}$, $\Gamma \in \mathbb{R}^{n_c \times n}$ must satisfy $\Gamma G^T = I_{n_c}$, and $G^T \Gamma = \tau$.

2.3.8 H_∞ Design Approach

A major emphasis in control design is the issue of robustness. As we have mentioned previously, uncertainties tend to exist in system models. In the case of a large flexible system, there are uncertainties in parameters such as dampings and frequencies of flexible modes. Also, models of flexible structures typically neglect dynamics of higher frequency modes. Our control objectives must be achieved despite these modeling errors. These two types of uncertainties can be described as structured and unstructured uncertainty. The control system must be robust enough to handle both types of uncertainty and still be able to keep the designed closed-loop performance. The control method described here is an H_∞ -based technique of “weighted gap optimization,” which is closely related to Glover-McFarlane loop-shaping.¹

Weighted Gap Optimization

The (weighted) gap metric is a measure of uncertainty based on the notion of the graph of a dynamical system. The primary motivation for this distance measure is

¹The H_∞ work presented here was performed by Scott Buddie for his M.S. thesis [84] under the guidance of Professor Malcolm Smith and was not directly supported by this contract. The work is included for completeness.

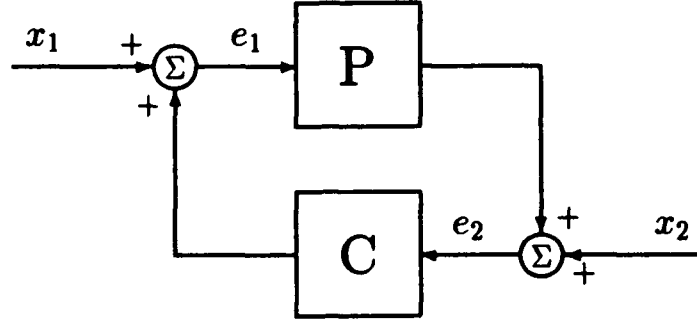


Figure 2.2: Standard Feedback Configuration

the need to handle unstable and marginally stable systems in a satisfactory way from the point of view of uncertainty. It may be claimed that a graph-based distance measure has certain advantages in other situations also, such as when lightly damped modes are present, as is the case with flexible structures.

The gap between systems P and P_1 can be defined as

$$\delta(P, P_1) := \max\{\vec{\delta}(P, P_1), \vec{\delta}(P_1, P)\} \quad (2.121)$$

where the directed gap $\vec{\delta}(P, P_1)$ is given by

$$\vec{\delta}(P, P_1) = \inf_{Q \in H^\infty} \left\| \begin{pmatrix} N \\ M \end{pmatrix} - \begin{pmatrix} N_1 \\ M_1 \end{pmatrix} Q \right\|_\infty, \quad (2.122)$$

and $P = NM^{-1}$ and $P_1 = N_1M_1^{-1}$ are normalized right coprime factorizations of the system transfer functions [85]. For weighting functions $W_i, W_o \in H_\infty$ such that $W_i^{-1}, W_o^{-1} \in H_\infty$, the corresponding weighted gap metric $\delta(P, P_1, W_o, W_i)$ is defined as:

$$\delta(P, P_1, W_o, W_i) := \delta(W_o P W_i, W_o P_1 W_i). \quad (2.123)$$

By using these weighting functions, one can emphasize or de-emphasize differences between the two plants at desired frequencies [86].

Consider the feedback configuration of Figure 2.2 where P and C are linear systems. The configuration, denoted by $[P, C]$ is defined to be stable if the operators $x_i \rightarrow e_j$ for $i, j = 1, 2$ are bounded. This is equivalent to

$$\begin{pmatrix} I & -C \\ -P & I \end{pmatrix}^{-1} = \begin{pmatrix} (I - CP)^{-1} & C(I - PC)^{-1} \\ P(I - CP)^{-1} & (I - PC)^{-1} \end{pmatrix} \quad (2.124)$$

being in H_∞ .

Let $[P, C]$ be a stable feedback configuration and define

$$b(P, C, W_o, W_i) := \left\| \begin{pmatrix} W_i^{-1} \\ W_o P \end{pmatrix} (I - CP)^{-1} \begin{pmatrix} W_i & -CW_o^{-1} \end{pmatrix} \right\|_\infty^{-1} \quad (2.125)$$

as the maximal amount of uncertainty in the weighted gap that a feedback system can tolerate [87, 88].

The *optimal robustness radius* can be defined as

$$b_{\text{opt}}(P) := \max_{C \text{ stblz}} b(P, C, W_o, W_i).$$

or

$$b_{\text{opt}}(P) := \left(\inf_{C \text{ stblz}} \left\| \begin{pmatrix} W_i^{-1} \\ W_o P \end{pmatrix} (I - PC)^{-1} \begin{pmatrix} W_i & -CW_o^{-1} \end{pmatrix} \right\|_\infty \right)^{-1}. \quad (2.126)$$

The problem of weighted gap optimization reduces to solving this well-posed H_∞ -minimization problem [85, 89].

The problem of finding a controller that is optimally robust with respect to the gap metric is identical to the problem concerning robust stabilization under coprime fraction uncertainty. This implies that any technique used to find a controller for coprime fraction uncertainty can be used to solve the gap metric problem. A solution to the coprime fraction uncertainty problem is given by Glover and McFarlane [90, 91, 92].

In the work of McFarlane and Glover [92], a certain "loop shaping" design procedure is proposed. This procedure begins with an augmented plant transfer function $P_{\text{aug}} := W_o P W_i$ containing input and output "weighting" functions W_o and W_i . The weighting functions are selected on the basis of a desired loop shape for the control system. The controller is designed to give optimal robustness with respect to normalized coprime fraction perturbations of P_{aug} . We point out that this choice of design synthesis problem is precisely the same as the H_∞ -minimization problem given by Equation (2.126). This algorithm was primarily used in calculating the controller for the truss structure.

Safonov and Chiang claim that Moore's reduced model can be constructed without computing a balancing transformation. This can be done by constructing projections

in terms of bases for the left and right eigenspaces of the product of the controllability and observability grammians $\Sigma_c \Sigma_o$ [93]. There are two methods which may be used to perform the first step of this algorithm which computes matrices whose columns form bases for the left and right eigenspaces of $\Sigma_c \Sigma_o$ associated with the "big" eigenvalues. The first method employs the Schur form of the matrix $\Sigma_c \Sigma_o$ and a Givens rotation to produce ordered values along the diagonal of the matrix. The second method is based on using the "square-roots" of Σ_c and Σ_o . We chose to use the first method. The decision to truncate modes is made by choosing the desired number of columns of the matrices whose columns form the bases for the left and right eigenspaces. It is important to note that the resulting transformed system is not a balanced realization, but it is an equivalent realization of the balanced realization obtained by Moore's algorithm. Therefore, if a system needs to be balanced, this approach cannot be used. Our concern in this effort was to avoid numerical conditioning problems, so an equivalent realization was deemed sufficient.

3. EXPERIMENTAL HARDWARE CONFIGURATION

3.1 Truss Structure, Controllers and Actuators

The structure is a 12-meter truss [43, 44] oriented in the vertical cantilever position, as shown in Figures 3.1 and 3.2. It is composed of four sections with 4 truss bays each. These four sections are bolted together to form the truss with a total of 16 bays. The truss is constructed from square aluminum tubing, with cross members of Lexan plastic bolted into place.

The active damping is achieved through eight momentum exchange proof-mass actuators. These devices are essentially linear DC motors with permanent magnets mounted on low friction sliding shafts and coils fixed to the housings as shown in Figures 3.3 and 3.4. The proof mass is centered with lightweight springs. Each actuator is supplied by its own current driver, which also incorporates viscous damping via a linear velocity transducer (LVT) mounted between the proof mass and the actuator housing. The actuators are capable of supplying up to 1 pound force from 2 Hz to 100 Hz. The locations of the actuators are shown in Figure 3.5. There are pairs of actuators mounted perpendicular to each other at the center of both the one-half and three-quarter stations on the truss as shown in Figure 3.6. These actuators affect only the bending modes, not the torsional modes. There are four actuators at the tip, two in each direction, mounted off center as shown in Figure 3.7. This orientation is used to affect both torsion and bending. Actuators can be monitored via three video cameras located on scaffolding beside the truss. The scaffolding also provides easy access to any of the actuator stations. The overall system is shown in Figure 3.8. Sensors for the control system are accelerometers, which are mounted on the actuator housings. The acceleration signals are integrated using bi-quad integrators with a break frequency of 0.1 Hz to provide velocity signals. A point light source is located at the tip—24 inches from the center of the truss. A digital camera can record the position of the light source, and thus provide information regarding the displacement of the tip.

During testing, the truss is excited via two alternative sources. The first alternative is an additional proof-mass actuator, identical to the control actuators. This adds dynamics to the structure which have been taken into account in the model of the structure. The second alternative is a large shaker, mounted to the wall at the tip of the truss. The shaker applies a force to the truss through a light connecting arm, and is capable of delivering forces up to 10 lb.

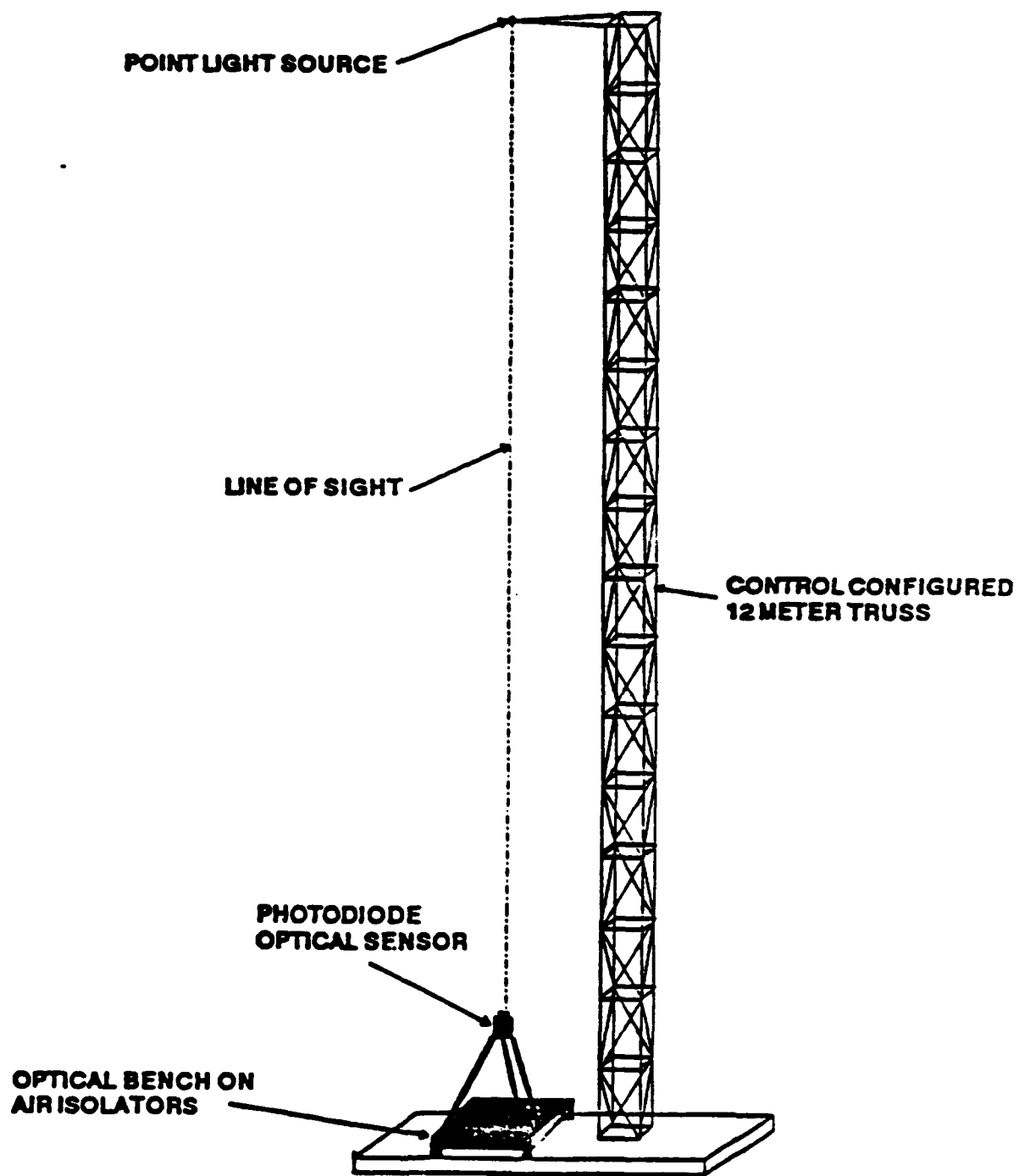


Figure 3.1: A 12-Meter Vertical Cantilever Truss Structure



Figure 3.2: 12-Meter Truss Experiment in Vibration Test Facility

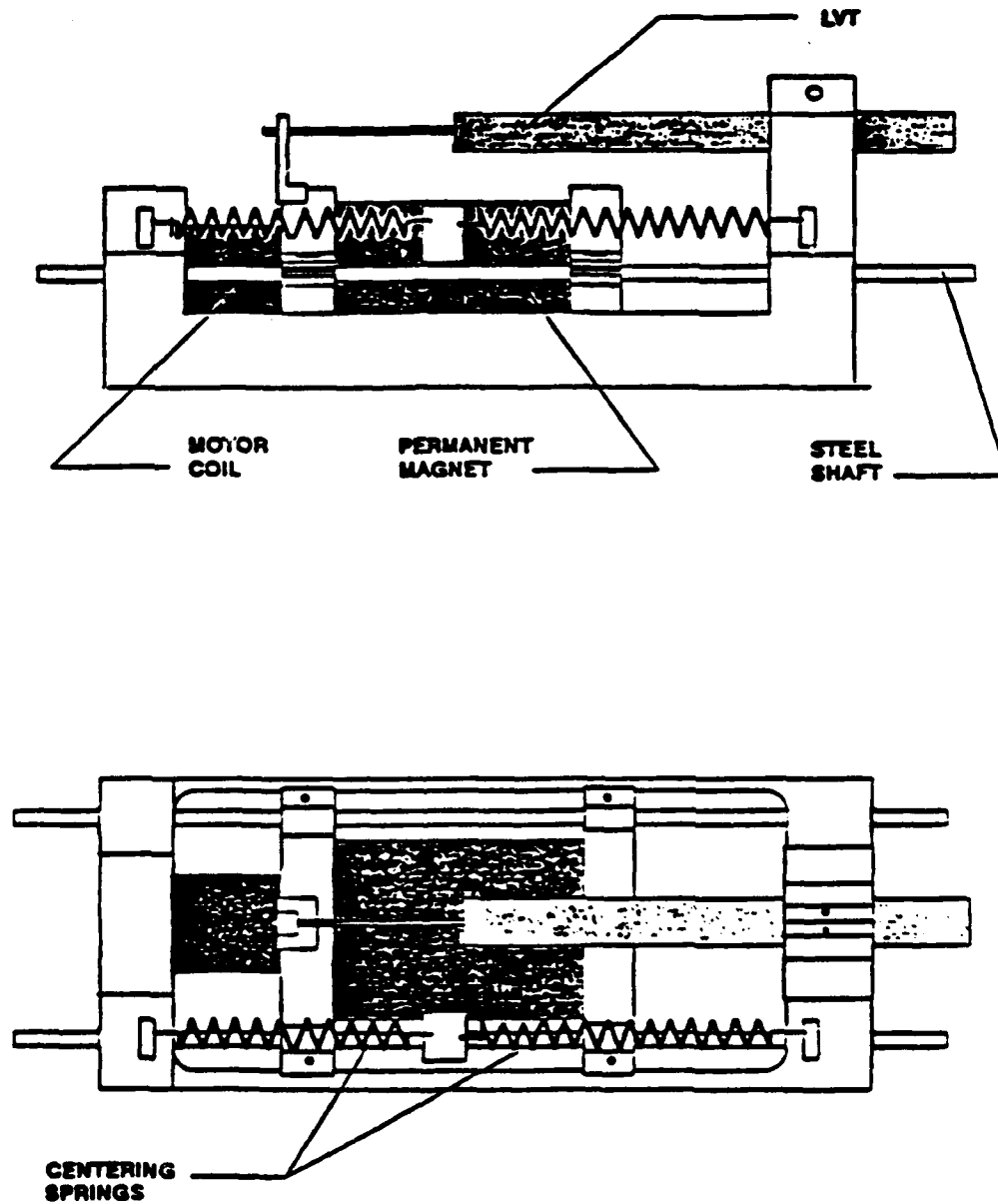


Figure 3.3: Momentum Exchange Actuator

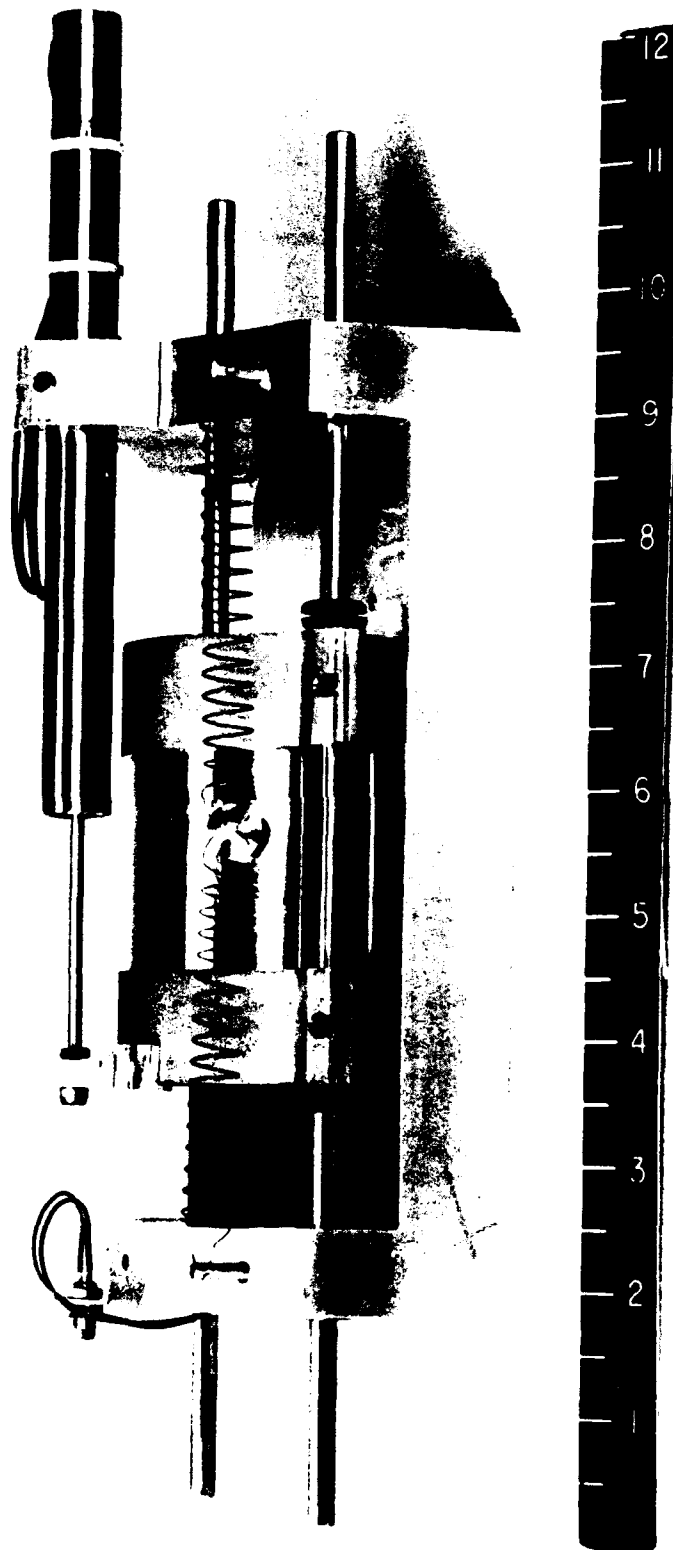


Figure 3.4: 12-Meter Truss Actuator

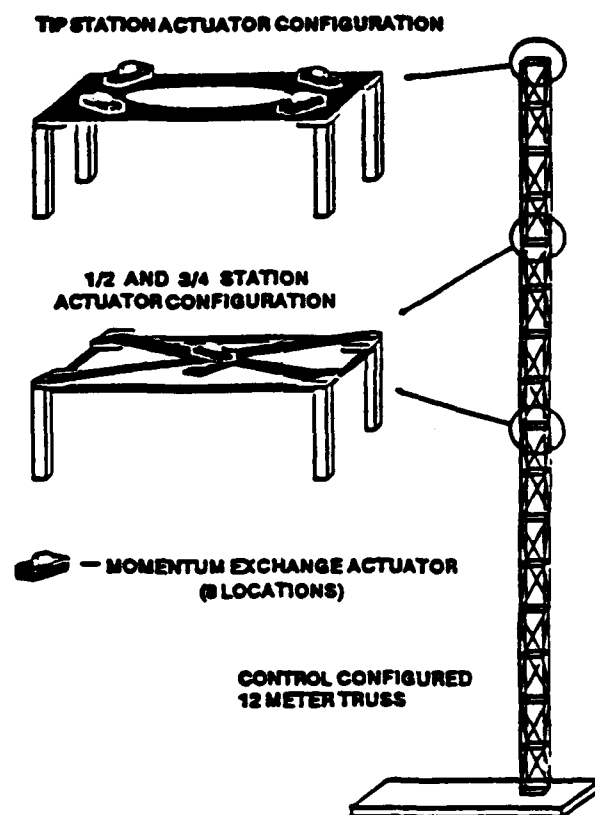


Figure 3.5: Actuator Locations on Truss



Figure 3.6: 12-Meter Truss 1/2 Station Actuators

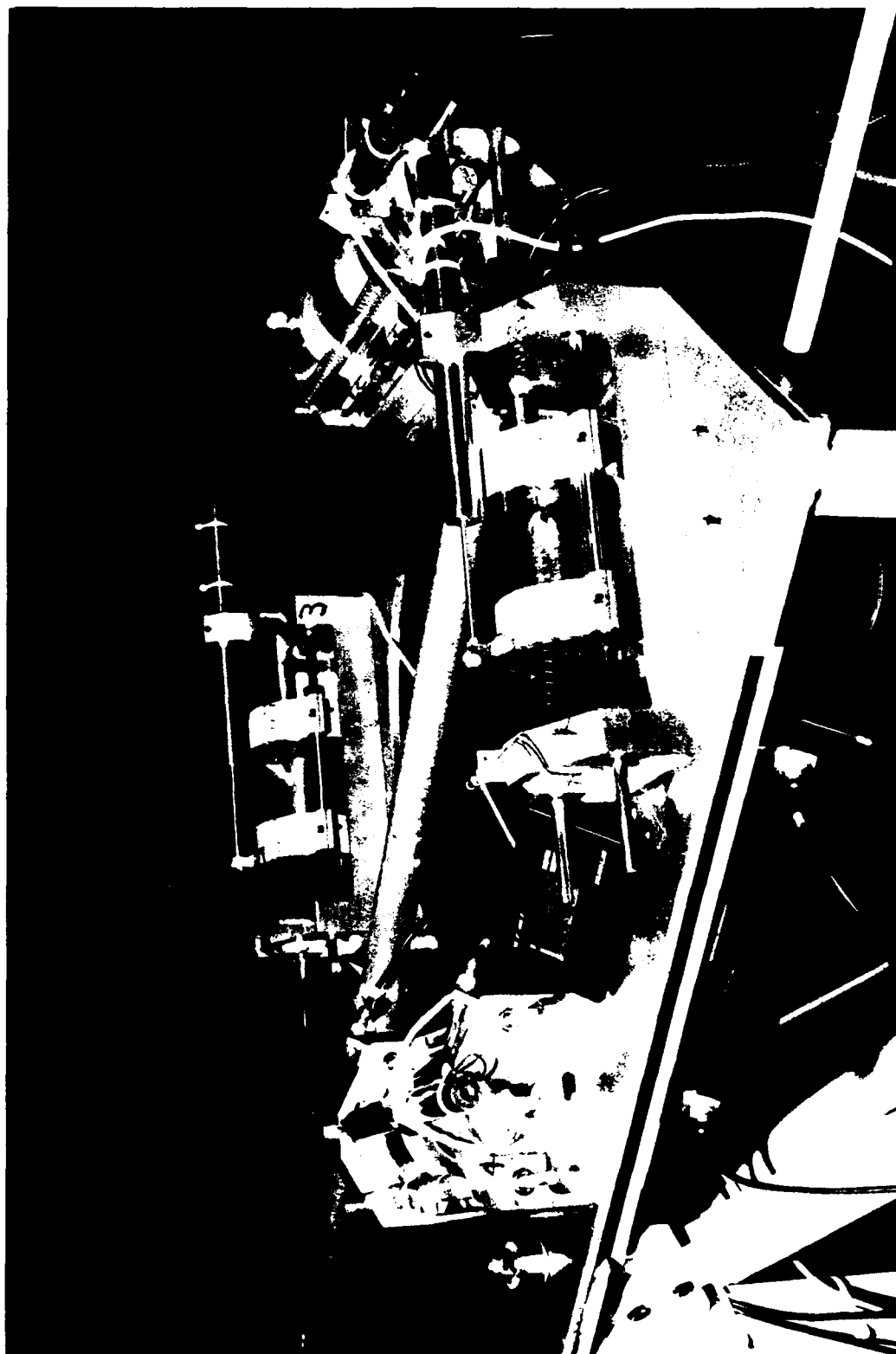


Figure 3.7: 12-Meter Truss Tip Station Actuators

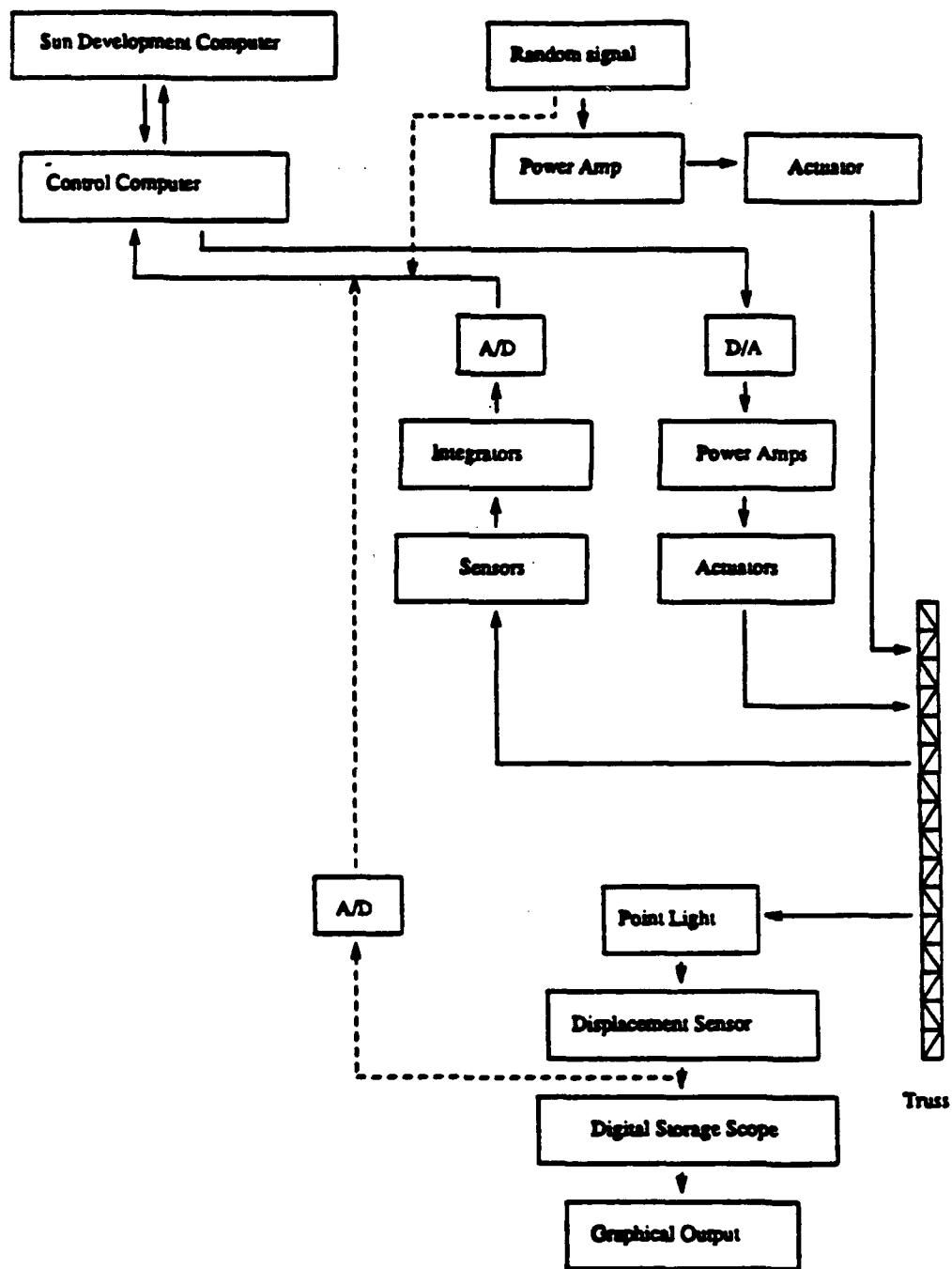


Figure 3.8: System Overview

3.2 Real-time Control System

The control algorithms are implemented digitally using a custom *Optima 3* control computer manufactured by Systolic Systems, Inc. The system consists of the control computer and a Sun 3/50 graphics workstation that serves as a development system. The control computer has a Motorola 68030 host processor with a 25 MHz clock and 4 MB of RAM. For high speed vector calculations, the *Optima 3* has a systolic array processor with a peak rating of 32 million floating point operations per second. External control is accomplished through a 12-channel, 16-bit, A/D and D/A card. The control programs are written in C programming language, compiled on the Sun workstation, and downloaded to the control computer. Unfortunately, the interface between the *Optima 3* and Sun is extremely slow.

Nevertheless, this system is capable of implementing rather complex control algorithms at sufficient frequencies for truss control. For example, a 14-state dynamic controller can easily be run at sampling periods less than 5 ms.

Data acquisition and analysis can be performed either by uploading data to the Sun host or through an OnoSokki dual channel spectrum analyzer. Since the interface between the control computer and the workstation is very slow, the spectrum analyzer is frequently used to minimize the time required to collect data for each experiment. Usually, rms values of the sensor and actuator signals were calculated on the *Optima 3* to take advantage of the array processing capabilities. The rms values were used as a quick performance summary of a given experiment for determining whether or not to save (upload) the entire data record.

4. SOFTWARE FOR CONTROL, SIMULATION, AND DATA ACQUISITION

4.1 Goals of the Software

One of the aims of the AF in-house project was to develop a real time computer control system which could be programmed in a higher level language. This would provide several benefits:

- Real-time control tasks such as reading A/Ds, controlling sampling period, etc. are all performed in easy to access, prewritten subroutines.
- Controller implementation is made easier—control laws can be performed in matrix form.
- Ability to store data during experiment, then upload for post processing.
- User friendly software possible.

As a result, the control and simulation software written can be used by an individual with minimal experience with the computer system. By the same token, additional controllers could be added to the software with minimal understanding of C programming language and specific knowledge of the Sun and Optima 3 computers.

4.2 Control/Simulation Program

Both control and simulation algorithms, including several different types of controllers, are implemented entirely in one piece of software. This was done for a number of reasons. First, we wanted to be able to test the controller software as thoroughly as possible without actually controlling the truss or running the software on the Optima 3. This enables the user to write, compile, and test the software on any Sun workstation. Second, it was felt that it would save effort in the end, since changes to the controllers or program features would only have to be made to one program. Third, it also gives the benefit of being able to run simulations on the Optima 3, which greatly decreases the run time.

4.2.1 Software Overview

The control/simulation software is interactive—prompting the user for various parameters about the run. A sample run of the program is given in Appendix A. The

prompts should be self explanatory. There has been some effort to make the program accept only correct responses, although it is certainly possible to enter erroneous data. Throughout the run, the program reports various events to the user each preceded by an ellipsis (...). These reports are merely intended as reminders of what is happening and as an aid for debugging.

The software has been divided into five files to make editing a bit more manageable.

exp.c This is the main program, which contains all the prompts to the user and performs all the calculations for control.

subroutines.c This file contains several important subroutines. The **get_matrix** routine loads ASCII files containing floating point numbers into array variables. The **mvm** routine multiplies a matrix and a vector. This routine uses the array processor when compiled for the Optima 3 and is one of the most important routines for implementing controllers.

band.white.fir.c This routine generates a bandwidth limited white noise signal. The routine uses the **rand** command to generate random short integers which are summed to produce a Gaussian distribution. The resulting signal is then filtered with a Finite Impulse Response (FIR) filter to limit the bandwidth to 50 Hz. If another cutoff is desired, another FIR filter could easily be designed using MATLAB. The filter changes the RMS value of the signal. To compensate for this, the standard deviation of the signal is calculated after filtering and then corrected to the desired value.

sin_disturbance.c This routine generates a sinusoidal disturbance and has the same form as **band.white.fir.c**

sin_sweep.c This routine generates a sinusoidal disturbance with a frequency which "sweeps" over some user-specified frequency range. This routine is similar to **sin_disturbance.c**.

There is one more file, **dummy_main.c**, which is used to get around a programming problem with the A/D and D/A routines provided by Systolic. The A/D and D/A routines apparently need to be aligned on 16-bit memory words instead of 8-bit memory words; if they are placed after other routines in memory, the compiler may not align them correctly. In C, the **main** routine must appear first. To get around this, a false or "dummy" **main** is used in **dummy_main.c**. This false **main** appears first in memory, but merely calls the real **main**, contained in **exp.c**, which appears after the A/D and D/A routines.

Although the software may appear to be long, it is relatively straightforward. Comments appear throughout the code. The major tasks performed by the software are,

1. Appropriate matrices are loaded from data files
2. Memory is allocated for signal storage
3. Disturbance signal is pre-calculated
4. Sensor biases are checked (if controlling truss)
5. Control/Simulation loop run
6. RMS signal values calculated
7. .m file written, containing report on the run
8. .dat file written, containing various sampled data signals from truss

4.2.2 Computer Compatibility

The program is written in C, using several standard libraries and several special ones. For most common operations, the standard `std.l`, `stdio.h`, and `math.h` libraries are used. For matrix operations, the Mathpac routines provided by Systolic systems are used. Systolic has also provided routines to read from the A/Ds, write to the D/As, control the sampling period, and measure intervals of time. These routines are in an object file, `cvrt16.o`.

The program is compiled on a Sun, but it can be compiled to run on three different types of machines, using the following makefiles,

- Sun 3/50 or 3/60 — `makefile`
- Sun 3/260 — `make_phonon`
- Optima 3 — `make_optima`

Different versions of the `std.l`, `stdio.h`, and `math.h` libraries must be used when compiling for the Optima, and these libraries have been provided by Systolic. The Mathpac routines are also different, since they are written to run on the array processor. Only a small number of Mathpac routines are available to run on the array processor, namely `vadd`, `vsub`, `vmul`, `vdiv`, `vsadd`, `vsmul`, `vmsa`, and `vsabs`. These

routines are in the library `qc.1`. It would be possible to run all of the Mathpac routines on the Optima 3's host processor at a reduced speed instead of the array processor if the Mathpac source code was available.

Compiler directives are used to make the program compile with the correct libraries, calls to subroutines, etc. For example, if the program is not compiled to run on the Optima 3, the A/D and D/A routines are not included, since the Sun will not have this capability. When changing from one machine to another, all of the program object files must be erased. This will insure that all the files are recompiled to run on the correct system. If this is not done, the linker will not be able to find all the subroutines called.

4.2.3 Memory Usage and Allocation

The Optima 3 has two types of memory. There is 4 MB of main memory, which can be accessed only through the host processor. This memory is used to store the program and most of the variables. When running a C program on the host, variables are automatically placed in this memory as well as locations reserved with `malloc` and `calloc`. Unfortunately, there does not appear to be any way to check how much memory is used. If too much is requested, the program crashes. The executable program requires about 160 KB and memory is allocated for variables as it is requested by the user. This limits the amount of memory the program actually uses. Most of the memory is used to store the real-time signals measured during the experiment. The amount of memory available for storing real-time signals from the experiment allows at least 30 seconds of data at 10ms sampling. The usefulness of this storage is limited, however, since the data link between the Optima 3 and Sun host is slow.

The second type of memory is on the array processor card. It is the only memory that the array processor can read, although the host can read this memory also. This memory must be allocated with the `APalloc` command, which returns a pointer to a floating point number. All variables used in calculations on the array processor must be stored in variables allocated in this manner. Only 32-bit, floating point numbers may be used with the array processor.

4.2.4 Sampling Rate

The sampling rate of the controller is precisely controlled using the A/D and D/A routines provided by Systolic. The sampling rate is set when the initialization routine `CVT_INIT` is run. The routine `CVT_STRT` starts the sampling rate timer. The first A/D call and the first D/A call each start separate timers. When the software reaches

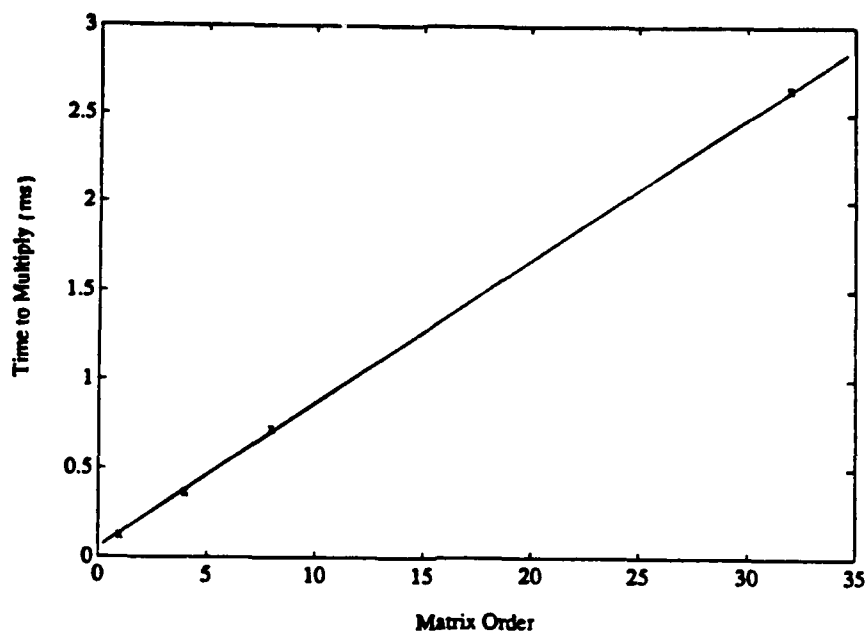


Figure 4.1: Speed of a Matrix/Vector Multiplication using Optima 3's Array Processor

the second A/D and D/A calls, the program waits until the proper time has elapsed since the previous calls. The program will stop and print an appropriate message if this sampling rate is exceeded. There is a two sampling period delay in the whole closed-loop cycle.

When compiled to run on the Optima 3, the program makes use of the array processor for matrix/vector multiplications and vector additions. For the multiplication, the Mathpac routine *vmsa* is used. It performs the dot product of two vectors. Due to the pipeline architecture of the array processor, this operation can be performed at virtually the same speed, regardless of the size of the vectors. This routine is called once for every row in the matrix, thus for square matrices, the speed of execution will be proportional to the order, rather than the order squared. Figure 4.1 shows the level of performance achieved.

It is easy to see that the matrix equations should be written with the smallest possible number of rows in the equations. Diagonal matrices can be multiplied with one *vmsa*.

The program as written can sample the A/D, write to the D/A, store all the signals and take care of all the other overhead in about 1.5 ms. A 14-state dynamic controller can be implemented with a sampling period of 5 ms. Perhaps some work could be done to reduce this.

4.2.5 Output of the Software

The time response data generated in the program is written to a data file in buffered binary output, which can be read directly by MATLAB. A MATLAB header is written to the file before any data to configure the file as a MATLAB `.mat` file [94]. The rate at which data points are written is selected by the user, thus the file size can be controlled. Data are written to the file one data block at a time using the C command `fwrite`. Using the `fwrite` command decreases the time required to upload data to approximately one third to one fifth the time required using `fprintf`. The size of the data blocks vary depending on the machine. The data written to the file may consist of any of the following: the time of the sampling instant, eight sensor outputs, eight controller outputs, nine mode displacements, force to shaker, disturbance signal and the LOS pointing error. Usually, some subset of these signals are written to the file because of the time required to upload the data. This file is placed in a subdirectory called `./data`.

In addition to the data file, a MATLAB command (`.m` file) file is written, containing information about the run. This file, in addition to the data file, is intended to be used with the post processor, which is described in Section 4.3. However, the `.m` also gives a good report of the rms values of the sensor signals, which can be printed as a record of the run or used for a quick analysis of the results.

4.2.6 Addition of New Controllers

If additional controllers are designed for the truss, whether by OSU or another party, there is no need to write additional software. New controllers can be added to the existing software relatively easily. Minimal knowledge of C programming is needed. A good reference is [95], since it was used by Systolic Systems Inc. when they wrote the C language routines for the Optima 3. It is also very helpful to read the documentation files written by Systolic, found in `/usr/download`.

The controller should be written in matrix form, to enable use of the `mvm` and `vadd` routines which run on the array processor. The constant matrices should be written in ASCII files, one line per row (for readability only). Any format readable by the `%f` option of the C command `fscanf` is acceptable, but it is advisable to use exponential format and as many decimal places as possible. Each matrix should be in a separate file. These files can be generated directly from CtrlC (using the command `save var >filename -r -c -f -(1x, 1p5e15.7))` or MATLAB (using the command `save`, then `translate`). If the controller is dynamic, it must be discretized for the sampling period at which it is to be run before it is saved in these files.

The program needs modifications in the file `exp.c` only. It is easy to spot the necessary

modifications, by studying the implementation of various controllers already coded. Basically, there are five areas to modify:

1. Define pointers to new variables used in the control law, making sure they are not already defined. The variables used may be the same as in other control laws, since only one controller is run at a time, however, they cannot be defined twice.
2. Add controller to main menu, giving it a controller number, which will be its reference throughout the program.
3. Add a **case** to the **switch** statement that loads the controller files. Also, at this point, memory for the constant matrices, controller states, and any temporary variables is allocated. If the controller is dynamic, there may be a switch to load controllers discretized at different sampling rates. The sign of any matrices should be corrected so there are no subtractions in the matrix equations.
4. Similarly, a switch must be added to calculate the feedback signal in the control loop. This relates the plant output vector y to the control vector u . If possible, use the **mvm** and **vadd** routines only. Since these only have two operands and they cannot be nested, temporary variables may need to be used. Also remember that all of these variables must be allocated on the array processor using **APalloc**.
5. Add a **case** in the **switch** that prints the controller title to the **.m** file.

4.3 Post-Processor Routines

The post-processor is a menu driven plotting routine written especially for the simulation program in MATLAB command files. It is intended that anyone with even minimal knowledge about MATLAB be able to run the post-processor. The post-processor can bring information about the simulation from the C program and write it on the plots, eliminating possible mix up of files.

Unfortunately, the low speed of the interface between the Optima 3 and the Sun has severely limited the usefulness of this program. With a faster interface it would be more useful. In any case, the following sections provide a short summary of its capabilities.

4.3.1 Capabilities and Structure

The post-processor is menu driven, calling many individual subroutines for specific tasks. It can graph actuator signals, sensor signals, fast Fourier transforms of sensor

signals, pointing error and mode displacements. The user can view these graphs with an option to print, or print them out directly. Menu options exist to plot groups of graphs. Additional features could be added easily. A sample run of the post-processor is shown in Figure 4.2.

```
>> Sim_1
>> pp
Post-Processor for 12 m Truss Simulations

1) View / Plot Actuator Signals
2) View / Plot Sensor Signals
3) View / Plot Actuator Power
4) View / Plot FFT of Sensor Signals
5) View / Plot Mode Displacements
6) View / Plot RMS Pointing Error
7) Plot All Actuator Signals and Total Power
8) Plot All Sensor Signals
9) Plot All Actuator Signals and Sensor Signals
10) Plot All Mode Displacements
11) Plot Total Power and RMS Pointing Error
99) Exit to matlab - Load new data

Option ? 1

Enter Actuator # or a for all Actuators 1

... Plotting

Do you want hardcopy ? (y/n) n
```

Figure 4.2: Sample Run of Post-Processor

Each graph is labeled with the parameters that are passed from the simulation program. This is done by writing a file that will be executed by MATLAB as a command (.m file). This file must be executed prior to running the post-processor routines, as it also runs an initialization routine.

The program was originally written to plot all the sensor and actuator signals. It is clear, however, that uploading all the channels will take an unacceptable length of time. It may be useful to modify the control/simulation program to prompt the

user for which channels are to be saved. This information could be passed to the post-processor through the .m file.

5. SYSTEM MODELING AND CONTROLLER DESIGN FOR AFWAL TRUSS

The truss was modeled by WL and the model was given to OSU in the standard FEM form, with no reduction of model order. The model was then transformed from nodal coordinates to modal coordinates by simultaneously diagonalizing the mass and stiffness matrices. The model order was then reduced using balanced model reduction techniques as described in Sections 2.2.1 and 2.2.2. Controllers were then designed in continuous time for the reduced order models then discretized for implementation on the control computer. Designing the controllers in continuous time was desirable since the sampling rate was not known at the time the controllers were designed. Also, many of the design tools which were used were for continuous time systems. It appears that the designs did not suffer much from this, since the sampling rates were sufficiently high.

5.1 System Modeling

The FEM model of the truss was constructed using MSC/pal 2 software. The model finally used was an equivalent beam representation of the truss. The stiffness properties of the beam were derived from a single bay of the truss. Using a single bay limited the number of degrees of freedom yet allowed an accurate approximation of the behavior of the truss for the lower frequency modes of interest. The beam model divides the truss into 16 elements and was not difficult to match with open loop testing of the truss. The full model has 33 nodes and 136 degrees of freedom.

The eight control actuators and single disturbance actuator are modeled as simple springs, masses and dampers, as shown in Figure 5.1. They are constrained to move in a single direction with respect to the reference frame. In actuality, they should be constrained to move in one direction with respect to the truss; however, for small deviations in the truss, it can be assumed that this is equivalent.

5.1.1 Modeling of Control Forces and Sensor Signals

Control forces were added to the FEM model considering the action of the proof mass actuator. The force produced is not with respect to a stationary reference frame, as would be the case with a thruster or an actuator attached to the wall. A proof mass actuator produces an equal and opposite force on the proof mass and the body of the actuator, as indicated in Figure 5.1. Recall the FEM equation (2.20) is

$$M\ddot{z} + D\dot{z} + Kz = Q \quad (5.1)$$

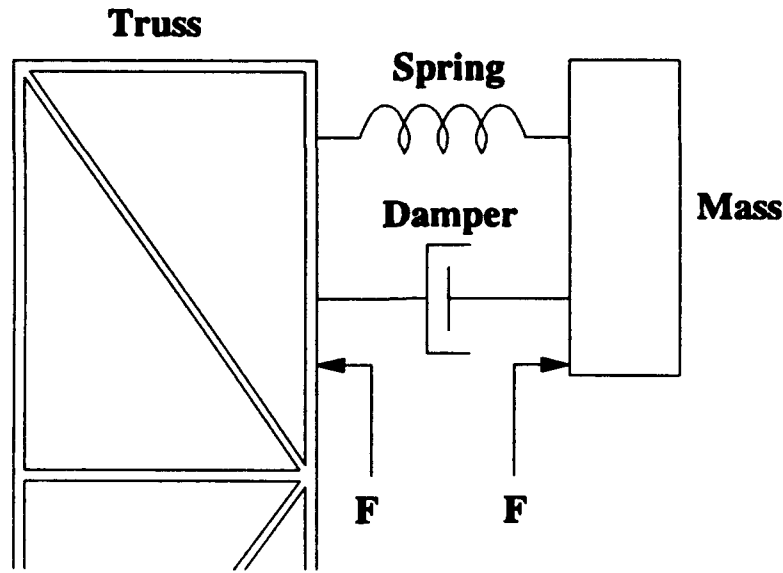


Figure 5.1: Model of Proof Mass Actuator

where

- z — nodal states (displacements and velocities in each direction, at each node)
- M — mass matrix (symmetric)
- D — damping matrix (symmetric)
- K — stiffness matrix (symmetric)
- Q — generalized external forces applied to each node, in each direction

Define the input to the actuators to be u and the output of the velocity sensors to be y , where the sensors are numbered as shown in Figure 5.2. The u and y are directly related to the nodal states,

$$y = C_q z \quad (5.2)$$

and

$$Q = B_q u \quad (5.3)$$

C_q has as many rows as there are outputs, and B_q as many columns as inputs. For point sensors that senses velocity along a single coordinate axis, each row of C_q contains a 1 at the nodal state where the sensor is located, and the rest of the elements in the row are zero. For momentum exchange actuators mounted along a single coordinate axis, each column contains a +1 at the nodal state where the

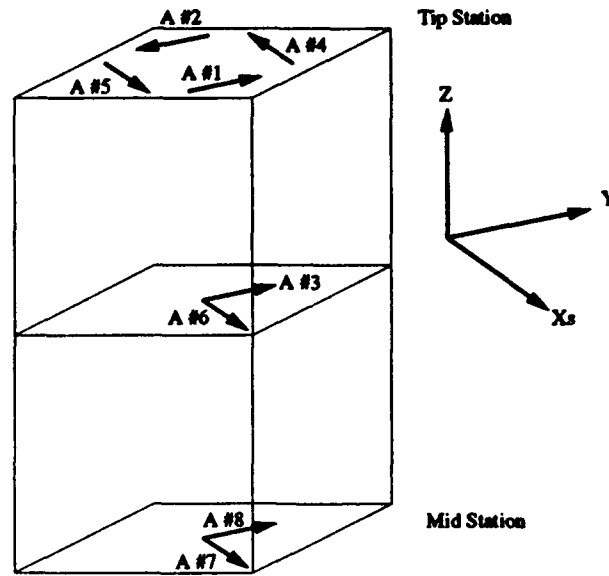


Figure 5.2: Sensor and Actuator Numbering for Controller Designs

actuator is attached the truss, and a -1 at the nodal state of the proof mass. If relative velocity sensors are used (such as the LVT sensors), it is clear that $C_q = B_q^T$.

5.1.2 Transformation from Nodal Coordinates to Modal Coordinates

The MSC/pal 2 software produces a FEM model with 272 nodal states. This model is transformed from nodal coordinates to modal coordinates as discussed in Section 2.2.1.

The FEM model produced by MSC/pal 2 software contains 10% damping in each proof mass actuator mode due to LVT feedback, but includes no structural damping. The structural damping is very light and hard to model. Dampings of 3% for the first bending modes and 1% for all other structural modes are assumed. This damping is added directly to the model in modal coordinates by letting

$$A = \begin{bmatrix} 0 & I \\ -\Omega^2 & -\phi^T D \phi - 2\zeta_s \Omega \end{bmatrix} \quad (5.4)$$

where ζ_s is the diagonal matrix of damping ratios and Ω is the diagonal matrix of squared modal frequencies.

Table 5.1: Structural Modes and Dampings — Open Loop, 10% Actuator Damping

Mode	Controlled					Uncontrolled					
	1st x	1st y	1st T	2nd x	2nd y	3rd x	3rd y	2nd T	4th x	4th y	3rd T
Frequency	1.75	1.75	6.64	8.46	8.45	19.42	19.40	20.22	29.35	29.31	33.90
% Damping	1.70	1.70	0.12	0.12	0.12	0.04	0.04	0.04	0.02	0.02	0.02

Table 5.2: Structural Modes and Dampings — Open Loop, 50% Actuator Damping

Mode	Controlled					Uncontrolled					
	1st x	1st y	1st T	2nd x	2nd y	3rd x	3rd y	2nd T	4th x	4th y	3rd T
Frequency	1.67	1.67	6.64	8.44	8.45	19.39	19.42	20.22	29.31	29.35	33.90
% Damping	6.32	6.33	1.08	1.07	1.07	0.67	0.68	0.69	0.62	0.62	0.61

5.1.3 Model Order Reduction

The modal transformation described in the last section results in a system equal in order to the FEM model, i.e. a model containing 136 modes. As predicted in Section 2.2.2, this model has too many modes to be used for controller design and the higher order modes are inaccurate. Therefore, the model was reduced as discussed in Section 2.2.2. Two models were created, one containing 14 states and another containing 42 states.

The controller designs for the truss were carried out using a 14-state model, including 5 structural modes and 2 actuator modes. A common problem with reduced order controller designs is the destabilization of uncontrolled modes. To test for this condition, the 42 state model was used as a higher order "truth" model to calculate closed-loop damping and to simulate the response of the truss. This model included 11 structural modes and all 10 actuator modes (8 control actuator modes and 2 modes to model the disturbance actuator which is oriented at a 45° angle to the 2 bending directions). The frequencies and natural dampings of the structural modes included in the truth model are shown in Table 5.1. The actuator modes are all approximately 1 Hz, with 10% damping.

Table 5.2 shows the frequencies and natural dampings of the structural modes included in the truth model for the case of 50% actuator damping.

5.1.4 Effects of Discretization and Sampling Delay

In the simulation of the truss response, the plant was discretized and the discrete-time response calculated. The discretized plant model and discrete-time simulations produced very similar results to those of the continuous-time model and continuous-time simulations. Thus, discretization did not present any problems.

However, a more serious problem was encountered with delay in the feedback loop.

The Optima 3, due to the way in which the A/D and D/A cards function, has a *two sampling period delay* between output and input. This delay was not taken into account when the controllers were designed, although it is modeled in the simulation routine. Several of the controllers have been destabilized, or nearly destabilized by this delay.

5.2 Controller Design

5.2.1 Performance Requirement—Tip Pointing Error

Some measure of merit was needed to judge the performance of the controllers, as well as actually design them. For this criterion, we chose to minimize the displacement of a point at the tip of the truss (24 inches from the center). The light source is located at this point, so the performance may be directly measured during experiments with the optical sensor as a Line-of-Sight (LOS) error. The distance of this point from the center of the truss was chosen to include the torsional modes of the truss.

The error term ρ can be written in terms of displacements of the top four sensors (y_1, y_2, y_4 and y_5).

$$\rho^2 = \left[\frac{1}{2}(y_1 + y_2) + \frac{12}{7} \sin 45^\circ (y_1 - y_2) \right]^2 + \left[\frac{1}{2}(y_4 + y_5) + \frac{12}{7} \sin 45^\circ (y_4 - y_5) \right]^2 \quad (5.5)$$

This can be rewritten in matrix form —

$$\rho^2 = y_d^T Q' y_d \quad (5.6)$$

where y_d is the vector of displacements at the sensors,

$$Q' = \begin{bmatrix} \beta & \xi & 0 & 0 & 0 & 0 & 0 & 0 \\ \xi & \beta' & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & \beta' & \xi & 0 & 0 & 0 \\ 0 & 0 & 0 & \xi & \beta & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (5.7)$$

and

$$\beta = 2.932 \quad \beta' = 0.507 \quad \xi = -1.219 \quad (5.8)$$

Now, let

$$y = C_d x, \quad (5.9)$$

where C_d is the output matrix yielding displacements at the sensor locations. This implies

$$\rho^2 = x^T C_d^T Q' C_d x = x^T Q x \quad (5.10)$$

This LOS error term was used in the design of optimal controllers.

5.2.2 Controller Design Techniques

This project presented the opportunity to test a wide variety of control techniques. Most of these were based on the Linear Quadratic Regulator. The cost function minimized in these techniques was

$$J = \int_0^\infty [x^T Q x + u^T R u] dt \quad (5.11)$$

or some variation thereof. Most of the controllers were designed using the LOS requirement described in the previous section as part of the cost criterion. The inputs were usually weighted equally. Thus, we typically chose $R = \alpha I$, where α is a proportionality constant used to control the ratio of the relative weighting of the states to weighting of the inputs (which is equivalent to the performance/control effort trade-off).

It should be noted that the limiting factor for closed-loop stability was typically travel of the proof-masses on the control actuators. In the controller designs, the gains grew larger as the states were weighted more heavily (or equivalently, the inputs weighted less heavily) to increase performance. The gains grew larger which required more force and, hence, further travel of the actuators. Eventually, a point was reached where the high gains demanded the proof-masses of the control actuators to exceed the maximum possible travel. Due to the travel limitations of the proof-mass, nonlinearities were introduced which resulted in instability.

Because the amount of travel required to create a given force decreases as the frequency increases, more control authority is actually available for modes above the first bending mode. Rather than using the LOS requirement in the cost function, these higher-frequency modes could be weighted more heavily—thus increasing the performance in the controlled higher-frequency modes. Although a few initial controllers were designed using equally weighted states, we mainly used the LOS requirement in

the cost function and did no *ad hoc* weighting in our designs. However, it should be noted that our results could perhaps be improved with such *ad hoc* weighting.

All of the following design techniques have been discussed thoroughly in Section 2.3 and in other sources [96]; therefore, only a short description will be given here.

Linear Quadratic Gaussian (LQG)

This approach is the standard full state feedback, using a Kalman filter to estimate the modal states [50]. The noise in each input and output is assumed to be uncorrelated and equal, so W and V are proportional to the identity matrix. The relative levels of these noises are unknown, so the relative weight is adjusted until the response of the observer is satisfactory. This is achieved with $4W = V$.

The Riccati equations were solved using MATLAB software [94]. If the weighting matrix Q was positive semi-definite, it was usually necessary to add a small number to the diagonal for numerical stability.

Linear Quadratic Gaussian with Loop Transfer Recovery (LTR)

This approach combines the frequency domain and state space techniques for a minimal phase system and is very similar to the LQG design discussed above.

Centralized Optimal Output Feedback (COFB)

This design simply minimizes the cost function given in the decentralized case, except there is no partitioning into subsystems. Again, DOLORES was used except in this case it was allowed to vary all the terms in the gain matrix. The feedback is of the form $u = Ky$, where K is not block diagonal.

Decentralized Optimal Output Feedback (DOFB)

This decentralized design is based on the following four groups of actuator/sensor (input/output) pairs: $\{1, 2, 3\}$, $\{4, 5, 6\}$, $\{7\}$ and $\{8\}$. The feedback is $u = Ky$ where K is 8×8 , u is the control vector and y is the vector of sensor velocities. K is block diagonal, where the blocks are 3×3 , 3×3 , 1×1 and 1×1 . This design, like all the other output feedback designs, was obtained with software called DOLORES which was developed at OSU to solve the optimal output feedback problem [58, 97].

Decentralized Frequency Shaping (DFS)

This design uses the same decentralized groups as the regular decentralized case. The cost function, however, consists of the plant outputs rather than the states [59, 96, 98]. The weighting of the control outputs was related to frequency, in order to discourage the controller from acting at high frequencies. The DOLORES software also incorporates a feature to augment states to the system and design these frequency shaping controllers. For first order filters, the resulting controller is an 8th order dynamic controller of the form $u = K_1 y + K_2 z$ where $\dot{z} = Fz + Gu$. K_1 and K_2 are block diagonal, partitioned as in the decentralized case, and F and G are diagonal.

Overlapping Decomposition (OD)

This method divides the truss into six disjoint subsystems, with the actuator/sensor (input/output) pairs grouped as $\{1, 2\}$, $\{3\}$, $\{4, 5\}$, $\{6\}$, $\{7\}$ and $\{8\}$. Each subsystem was minimized separately using DOLORES. The feedback is $u = Ky$ where K is 8×8 , u is the control vector and y is the vector of sensor velocities. K is block diagonal, where the blocks are 2×2 , 1×1 , 2×2 , 1×1 , 1×1 and 1×1 .

Controlled Component Synthesis (CCS)

For CCS controller design, the full 136 degree of freedom FEM of the truss was decomposed into two component models. The lower 10 bays comprised the lower component which had 62 degrees of freedom. The upper 6 bays comprised the upper component which had 80 degrees of freedom. These component models were formed from the composite system model using the procedure discussed in Section 2.3.6 since the composite model was available.

Modal decomposition was performed on the FEM of each component by solving the appropriate eigenvalue problems for the component modes and modal frequencies. For numerical reasons, the lower and upper component models were reduced to 20 and 44 degrees of freedom, respectively, using frequency truncation. At this stage, all component modes below 100 Hz were retained. A state space model was then formed using the modal states. Taking the sensor velocity measurements as the outputs and the actuator signals as the inputs, the component models were further reduced based on observability and controllability of the modes. The resulting models were used for controller design.

Static optimal output feedback gains were found for each component so as to minimize motion at the internal boundary components (the one-half and three-quarter stations). The motion at the tip was also weighted to meet additional performance

objectives. A quadratic performance criterion as in Equation (2.110) was used for the component controller designs. The resulting static output feedback control law for each component was of the form

$$u = K_{css}y. \quad (5.12)$$

Optimal Projection (OP)

This approach uses stochastic modeling for the uncertainty of the system order to improve the robustness with respect to the unmodeled higher frequency modes. The controller is of the form $u = K\hat{x}$ where $\dot{\hat{x}} = A_c\hat{x} + Fy$. The order of the controller is predetermined—in this case ten.

Decoupled Optimal Projection (DOP)

This method is a decentralized version of optimal projection. The controller is essentially the same, except three third order controllers run independently, one for each bending direction and a third for the torsion.

H_∞ Gap Metric Design

The goal in this method is to shape $(I - PC)^{-1}P$ for disturbance attenuation and $C(I - PC)^{-1}$ for shaping the control action required.¹ In this case, only the top four actuators are used for control. The multivariable design is a centralized design with the four sensor/actuator pairs as the inputs and outputs of the system.

A balanced truncation of the 42-state "truth-model" yields a suitable design model. The balancing transformation and truncation are performed with the algorithms discussed previously. The resulting design model from has 14 states corresponding to the first five structural modes and two actuator modes. The gap calculated between the modal truncated model and the balanced truncated model was found to be 0.02. The difference between the two models is primarily due to the actuator modes. Since actuator modes need to be considered for stability of the system, one might question whether or not two modes are enough. To answer this question, both reduced models were compared to a 30 state model which contained the same five structural modes and all 10 actuator modes. The gap between either reduced model and the 30 state model was found to be around 0.09. This value was judged to be acceptable; therefore, a fourteen state model seemed appropriate for design.

¹This portion of work was performed by Scott Buddie.

The weighting function $W_B = kW$ is applied to each channel. For the constant weight, W is set equal to I and k is varied to see what happens to the resulting closed loop system. Initially, k was set to I to get an initial indication of the closed loop responses. By doing this, the modes were shifted by what appeared to be an excessive amount. After some additional trials, an acceptable range of k was found. The controllers were calculated using the four-block solution of Englehart and Smith [89] for reasons stated later. Therefore, b_{opt} was found from the four-block solution.

In case the 14-state controller could not be implemented at the desired sampling rate, controller reduction schemes were employed to generate 12-state controllers. An upper bound for the weighted controller gap between the full and reduced order compensators can be found by taking twice the sum of neglected singular values of the product of the controllability and observability matrices. This controller gap comes out to be about 0.16 for $k = 0.5$.

Since the optimal controller algorithm was causing the truth model to become unstable, we looked at introducing a first order weighting function to reduce control action in the higher frequency modes. In this multivariable case, we will introduce the first order weight into each of the channels; thus, creating a weighting function with 4 states. This tells us that a full order controller using this weighting scheme will consist of 21 or 22 states depending on the algorithm chosen. In this case, we have decided to use the optimal controller algorithm to see if the truth model could now be stabilized. The controller is reduced to 15 states for simulation and implementation. Two first order weights are chosen. The frequency chosen as the roll-off frequency was 7.162 Hz (45 rad/sec). This was chosen somewhat arbitrarily to give a little less weighting to the torsional mode and the 2nd x and y mode. The frequency could be shifted depending on the weighting desired. The difference in the first order weights is due to a constant term.

The controllers are discretized at a sampling period of 5 ms. This was determined to be a sufficient amount of time to process the control law and was fast enough in terms of the modes of the system that were looked at. In the implementation of these controllers, there is a two sampling period delay due to the A/D and D/A boards as mentioned in Section 4.2.4. This delay could cause our closed loop system to go unstable. The delay is of the form $e^{j\omega T}$ where T is 10 ms and ω corresponds to a modal frequency. Therefore, the higher the modal frequency, the greater the phase shift that occurs. To investigate this, the exponential is modeled by the function $\frac{1 - .005s}{1 + .005s}$. The function is then augmented to the plant and the eigenvalues of the closed loop system are compared to the eigenvalues of the plant without delay. The eigenvalues in all cases were hardly affected by the delay; therefore, controllers were designed for the plant—without the delay being considered. If the delay was determined to cause a problem, then one could augment the plant with a delay function and do the design based on the augmented plant.

5.2.3 Output Feedback and Actuator Damping

Through the course of the designing controllers, it was realized that there is a strong relation between the addition of structural damping and actuator damping. As more damping is added to the structural modes, the actuator modes become more lightly damped. This can be seen in the root locus for constant feedback of a single sensor and actuator pair. The structural and actuator modes move in different directions as the feedback is increased. This is true since the proof mass actuator gives the proof mass and the truss equal and opposite forces, and only the velocities of the truss are available for feedback.

There are at least two solutions. One solution is to feed back the velocities of the proof masses in addition to the velocities of the truss, or perhaps only the relative velocities. This could be done using the LVT sensors. There is another solution, however, which is simply to increase the natural damping of the actuator modes. Much more damping can be added to the structural modes before the actuator modes approach the $j\omega$ axis and, thus, instability.

6. SIMULATION AND EXPERIMENTAL RESULTS

In this chapter, results of the experiments on the truss are presented. In general, we have shown that the experimental behavior of the truss is very similar to the simulated behavior of the model, both in open and closed loop testing. Several controllers were tested using various inputs. Although some of these controllers did not adequately stabilize the truss, the results typically coincide with predicted behavior. As the H_∞ controller designs differ considerably from the LQR-based designs, the H_∞ results will be presented in a separate section from the other results.

6.1 Noise Levels of Velocity Signals

The level of noise in the velocity signals presents one of the first problems encountered in this experiment. As mentioned in Chapter 3, the signals from each accelerometer are integrated to provide velocity measurements using analog circuits built into the current drivers for each actuator. The analog circuit has a cutoff at 0.1 Hz, and zero gain at DC. Therefore, the frequency response achieves maximum magnitude at 0.1 Hz and rolls off at -20 dB/decade as the frequency increases. The noise from the accelerometers is assumed to be white, however, the integrators tend to amplify the low frequency noise to the point where it could be a problem.

Table 6.1 shows typical noise levels. These data were gathered through the Optima 3 while the truss was at rest and represent the A/D noise, noise in the integrator, and noise in the acceleration signal. The low frequency noise does not seem to cause significant problems for the controllers. Any noise which passes through the controllers is attenuated by the truss. The main problem which is caused by the noise is difficulty in measuring the performance of the truss, especially in terms of the steady-state velocity RMS values.

Obviously, reducing the noise in the signals would improve the quality of the data. Reducing the noise, however, would have required redesigning the integrators, which would have been costly and undesirable. Another option was to increase the disturbance signal to the truss. This was tried and achieved some measure of success. Perhaps the best long term solution would be to avoid using integrators which by nature amplify low frequency noise.

Table 6.1: Typical RMS Noise Levels of Velocity Signals (inch/s)

Sensor	#1	#2	#3	#4	#5	#6	#7	#8
Velocity	0.019	0.014	0.048	0.018	0.020	0.030	0.021	0.035

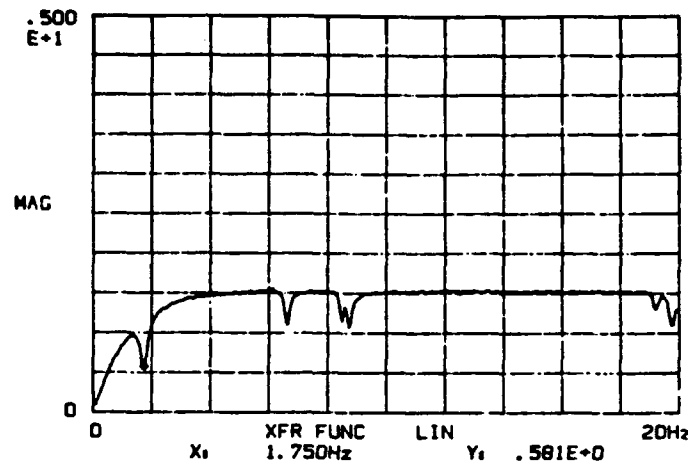


Figure 6.1: Bode Plot of Shaker with Voltage Drive

6.2 Disturbance Generation

As previously discussed, the truss can be excited from either the disturbance actuator which is mounted on the truss, or the shaker which is mounted to the wall and connected to the tip of the truss. Each of these disturbance generators presents advantages and disadvantages.

The disturbance actuator was limited in both the travel of the proof mass and the maximum force generated. The natural damping of the disturbance actuator could be increased to reduce the travel of the proof mass, but this would also change the model of the truss slightly. Even if the natural damping was increased, the maximum force which the actuator could achieve would still pose a problem, especially with random signals. The maximum output of the D/A converter was ± 10 V which converts to ± 1 lb. Thus, the RMS output of a random signal was limited to less than 0.4 lb. The maximum D/A output was indeed the limiting factor in determining the maximum force, not the power amp or the actuator coil. This problem could be solved by amplifying the analog signal, but a suitable amplifier could not be found.

The second disturbance generating alternative is the shaker mounted to the wall near the tip of the truss. This device has the potential to impart a much larger disturbance force to the truss, although the travel of the tip of the truss is limited to 1 inch peak-to-peak. At first, a voltage source was used to drive the shaker. This did not perform well at the resonant frequencies of the truss. Figure 6.1 shows the Bode plot of the shaker, from the voltage command to the output force, measured by the force gauge at the end of the shaker arm. The dips in the transfer function at the resonant frequencies are caused by the back EMF of the shaker. Driving the shaker with a current source eliminated this problem. The current driver for the disturbance

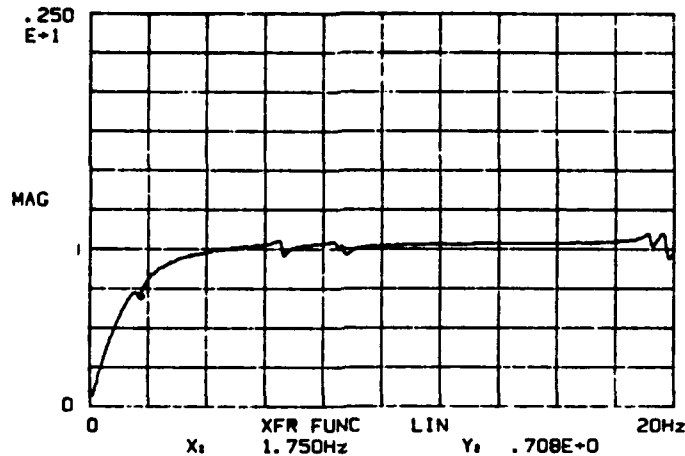


Figure 6.2: Bode Plot of Shaker with Current Drive

actuator was used to drive the shaker, and this appears to have worked moderately well, as shown by Figure 6.2. The response at low frequencies, however, remains a problem. Generating larger forces at lower frequencies would require more travel of the shaker arm. At the first bending mode, we can see that the output of the shaker is about 71% of the output at higher frequencies. This is due to the poor low frequency response of the shaker. In theory, this response should be flat. This shortcoming is magnified because most of the response of the truss is due to the first bending modes. This problem explains why some of the data do not match the simulations well as would be expected.

Perhaps the low frequency problems with the shaker could be corrected by prefiltering the disturbance signal with a filter having a transfer function which is approximately equal to the inverse transfer function of the shaker over the frequency range of interest. The filter could be constructed by curve fitting the desired transfer function over the frequency range of interest and using several MATLAB tools which would make this design easy. For example, the command `yulewalk` could provide a good approximation. Also, a model of the shaker could be determined and included in the simulations.

Another solution to this problem is to find a shaker that has a better low frequency response.

As previously discussed in Chapter 4, the disturbance signal is generated in the control/simulation program. Figure 6.3 shows the spectrum of this signal leaving the D/A as recorded with the OnoSokki. We chose to bandlimit the signal to 50 Hz which allows us to put more energy into the frequencies which contribute most to actually exciting the truss. This frequency range also includes several uncontrolled

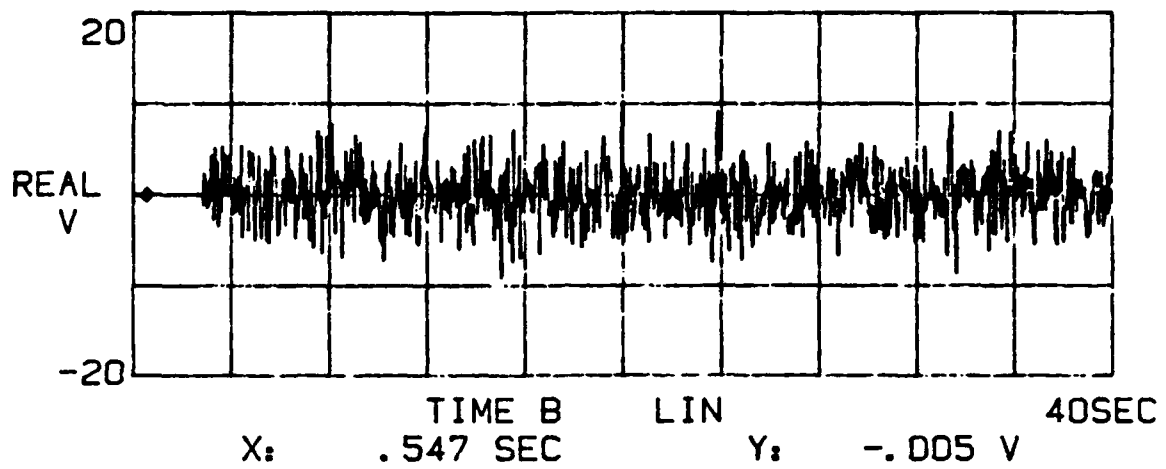


Figure 6.3: Bandwidth Limited Random Disturbance Signal

modes, so the results would allow the controllers to be tested for susceptibility to spillover.

6.3 Bode Plots

Bode plots of the structure can be generated relatively easily, using the OnoSokki dual channel spectrum frequency analyzer. The OnoSokki obtains both the magnitude and phase of the transfer function by taking the Fast Fourier Transform (FFT) of the input and output, and then dividing each frequency point. This process is repeated and averaged to obtain a smoother plot. The lower the frequency range, the longer it takes to obtain a smooth plot. For 0-50 Hz, a good plot can be obtained in about one minute. For 0-10 Hz, at least three minutes is necessary. This presents no problem for the control program, so long as the data are not stored in the Optima 3 for the entire length of the experiment.

Bode plots are also easily obtained from the data which are uploaded from the Optima 3 to the Sun Workstation. A MATLAB routine is used which implements basically the same method as the OnoSokki. The experiments were performed for more than 164 seconds to provide 16,384 data points per channel per experiment. These time data were used to compute 4096-point FFT's for 15 windows with 50% overlap. A Hanning window was used when computing the FFT's of each window. The frequency responses were calculated by taking the ratio of the FFT of the output to the FFT of the input. The resulting 15 responses were then averaged together to create a smoother plot.

Open loop plots for both experimental and simulated responses are shown in Figure 6.4. Agreement between the experimental and simulated results will be discussed in the next section. Only Bode plots for the first sensor output with respect to the disturbance input are shown. The Bode plots of other sensor outputs are very similar. Note that bending modes in the x and y directions are at virtually the same frequencies due to the symmetry of the truss.

Closed-loop Bode plots of the truss with several different controllers are shown in Figures 6.5 through 6.9. Each Bode plot shows the experimental and simulation magnitude comparisons. Figure 6.9 compares the closed loop frequency responses with the open loop response from experimental data. The differences between the closed loop responses can be hard to see, thus only a few are included. These Bode plots are perhaps not the best way to compare the controllers as the controllers were designed to minimize the motion of the point light source. The actual cost criteria which were minimized for the various controllers are, in some cases, theoretically and fundamentally different even though the objective of each controller design is the same. Regardless, these responses do provide significant insight.

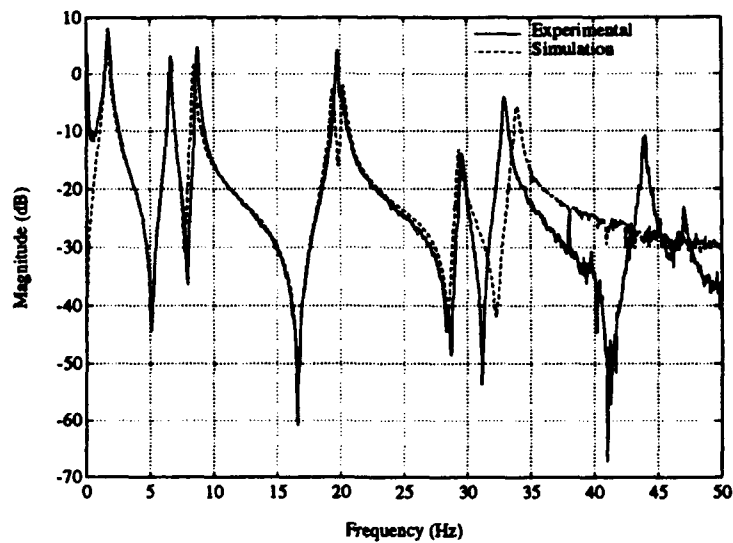


Figure 6.4: Comparison of Experimental and Simulation Open Loop Frequency Responses, Velocity #1, Shaker 1-lb RMS Random

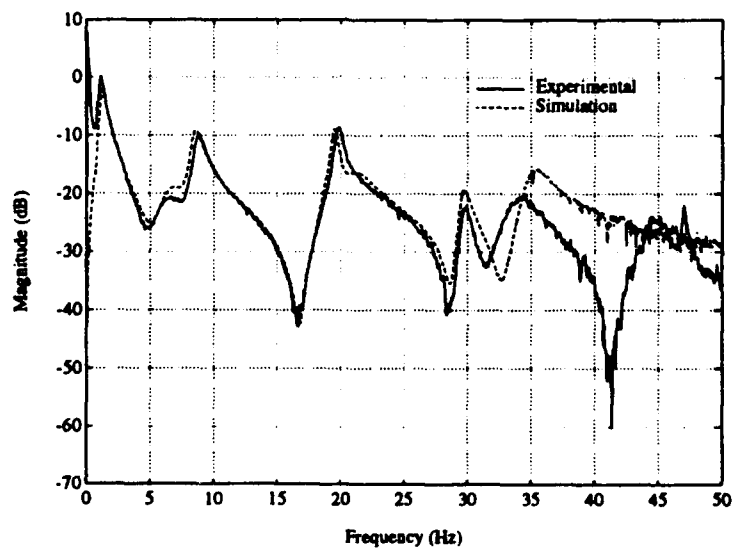


Figure 6.5: Comparison of Experimental and Simulation DOFB Frequency Responses, Velocity #1, Shaker 1-lb RMS Random

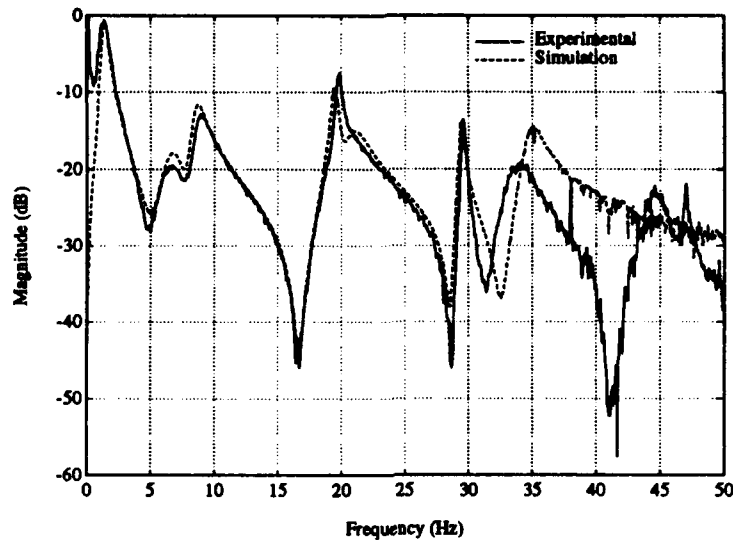


Figure 6.6: Comparison of Experimental and Simulation COFB Frequency Responses, Velocity #1, Shaker 1-lb RMS Random

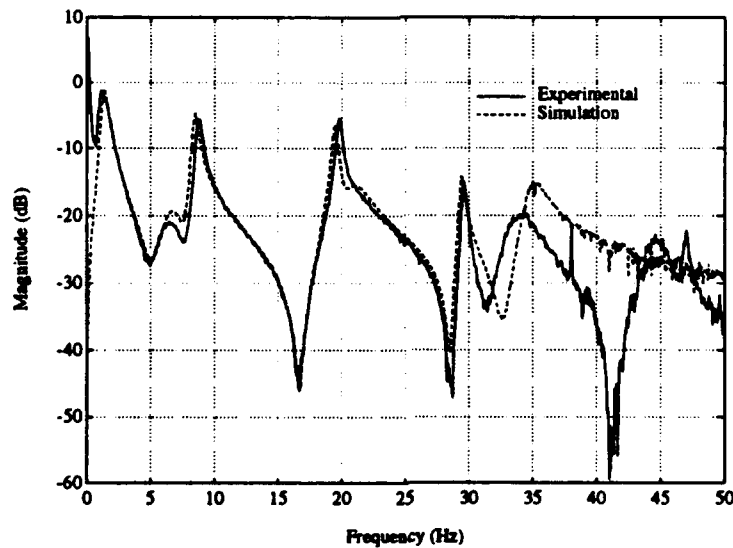


Figure 6.7: Comparison of Experimental and Simulation Overlapping Decomposition Frequency Responses, Velocity #1, Shaker 1 lb RMS Random

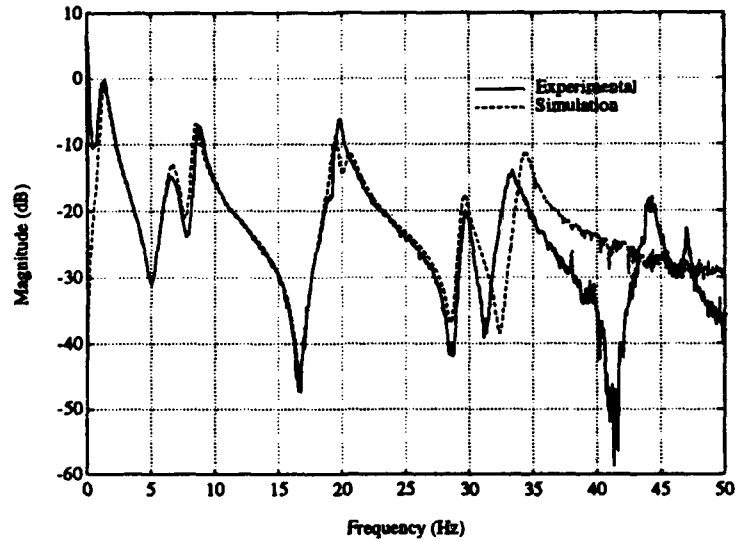


Figure 6.8: Comparison of Experimental and Simulation Controlled Component Synthesis Frequency Responses, Velocity #1, Shaker 1-lb RMS Random

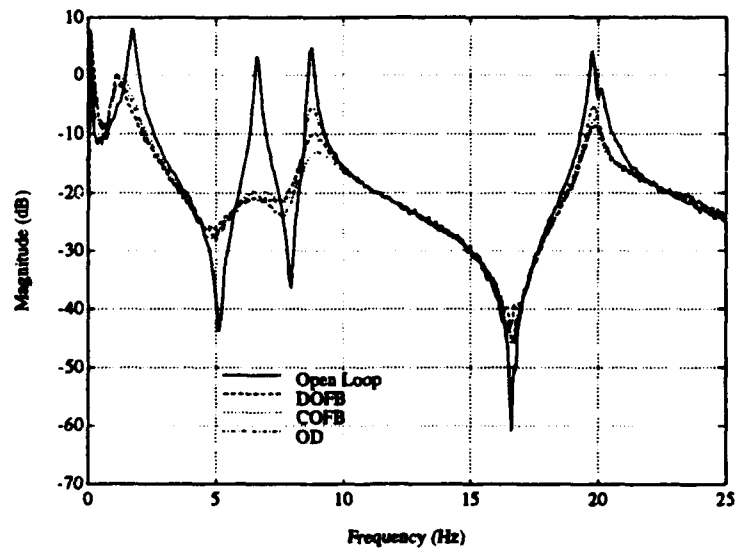


Figure 6.9: Comparison of Experimental Frequency Responses for Several Controllers, Velocity #1, Shaker 1-lb RMS Random

Frequency responses of the light source position from the disturbance input to the LOS sensors have also been generated from both experimental and simulated data. When the experimental data were taken, the gain of the amplifier for the photo array signals from the optical sensor was set incorrectly. Therefore, the amplitudes of the time response data were off by a factor of 10. Knowing the source of this error, the data were post-processed by multiplying the time signal by 10 or, equivalently, by adding 20 dB to the frequency response to give the correct results.

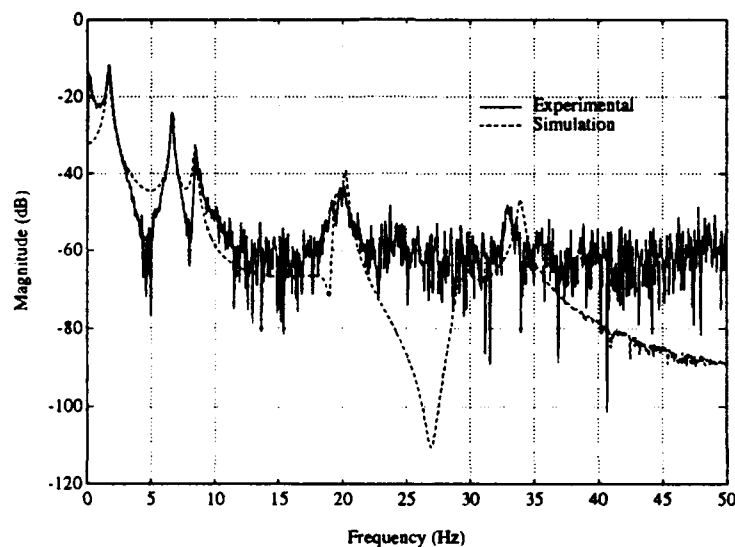


Figure 6.10: Comparison of Experimental and Simulation Open-Loop LOS Frequency Responses, LOS x , Shaker 1-lb RMS Random

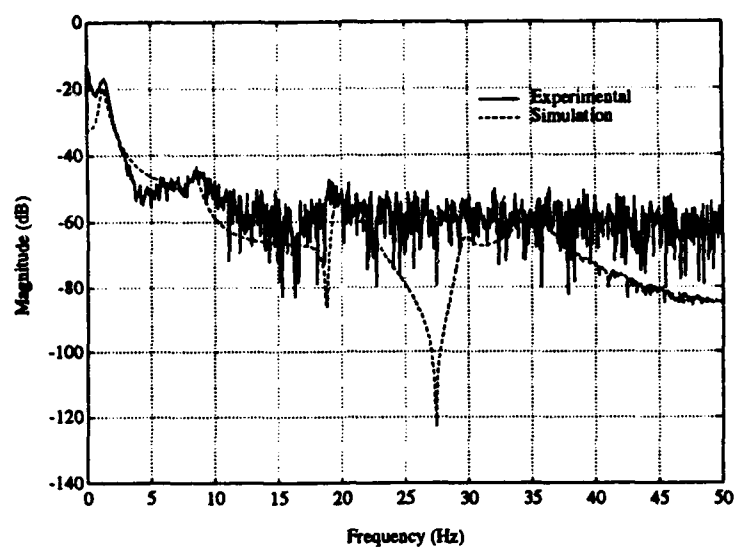


Figure 6.11: Comparison of Experimental and Simulation DOFB LOS Frequency Responses, LOS x , Shaker 1-lb RMS Random

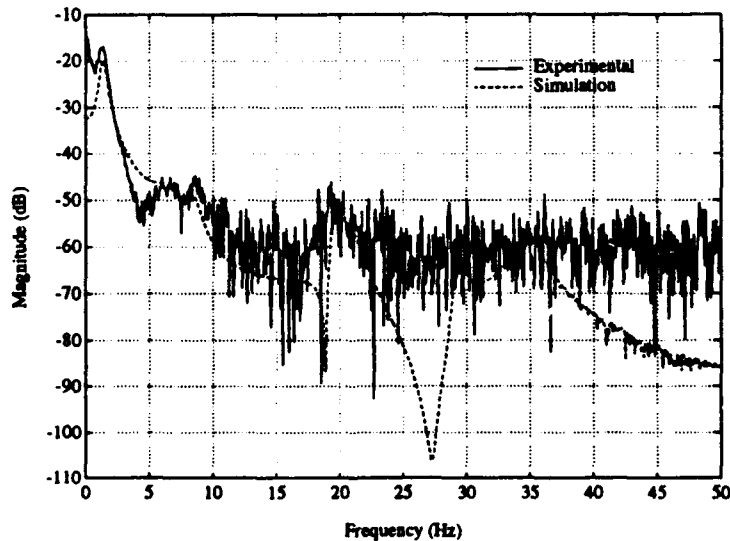


Figure 6.12: Comparison of Experimental and Simulation COFB LOS Frequency Responses, LOS x , Shaker 1-lb RMS Random

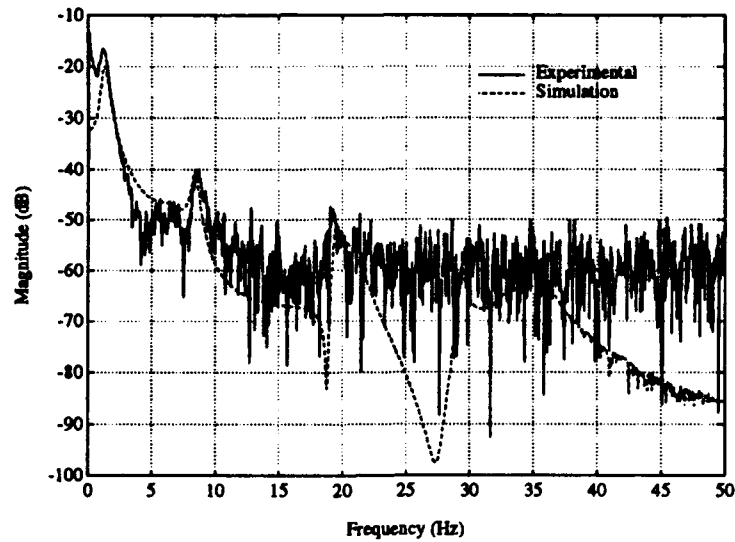


Figure 6.13: Comparison of Experimental and Simulation Overlapping Decomposition LOS Frequency Responses, LOS x , Shaker 1 lb RMS Random

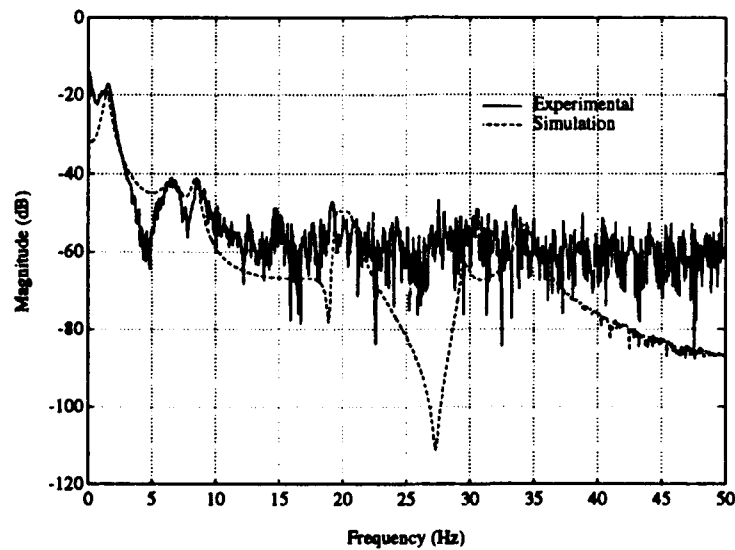


Figure 6.14: Comparison of Experimental and Simulation Controlled Component Synthesis LOS Frequency Responses, LOS x , Shaker 1-lb RMS Random

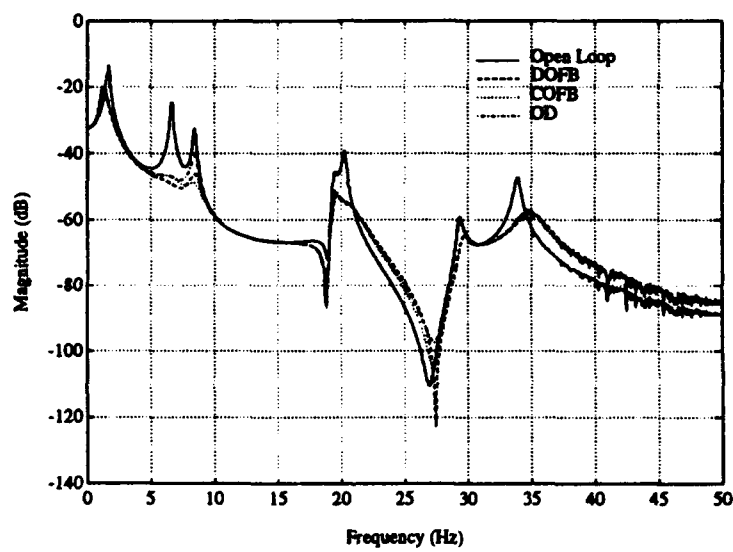


Figure 6.15: Comparison of Simulation LOS Frequency Responses for Several Controllers, LOS x , Shaker 1-lb RMS Random

Figures 6.10 through 6.14 compare the corrected experimental frequency responses of the LOS position in the x direction with the responses from simulation. Note that these frequency responses match very closely at the modal frequencies. Figure 6.15 compares the closed loop frequency responses and the open loop response from simulated data. The simulated data were used because the responses are smoother than the experimental responses and thus show the added damping more clearly. From Figures 6.10 through 6.14, one can see that the added damping shown here is essentially equivalent to the added damping from the experimental data. Again, the differences between the closed loop responses can be hard to see, thus only a few are included.

One problem with the post-processing technique described above is evident in these responses. Although the post-processing corrected the signal, it also amplified the noise to levels the noise never actually achieved. This is evident in the experimental responses where we see a noise floor of about -60 dB. It is assumed the actual noise present in the frequency response is due to the noise in the photo array or resolution (quantization) of the photo array and not due to the amplifier itself. If this is indeed the case, then had the amplifier been set correctly, the noise floor would have been about -80 dB. Thus, the frequency responses generated with the post-processed (corrected) data do not accurately depict the true noise floor.

6.4 RMS Velocities

The link between the Optima 3 and the Sun Workstation is very slow and the amount of data that can be uploaded is therefore limited. The RMS values of the sensor and actuator signals are very useful for evaluating controller performance and can be uploaded in less than ten seconds.

Table 6.2 shows the RMS sensor velocities and actuator power measured experimentally, with a 1-lb RMS random disturbance from the shaker. The last column is the total RMS of all the actuator signals. These data were taken for 164 seconds after the truss was allowed to reach steady state for 30 seconds. The same random number generator seed was used for all experimental trials.

Table 6.3 shows the simulated sensor velocities and actuator power for the same controllers. The simulation results are similar to the experimental results.

6.5 Modal Damping

One of the simplest yet perhaps most meaningful way to evaluate controller for a LSS is to compute the closed-loop damping. This provides a good indication of the

Table 6.2: Experimental RMS Sensor Velocities and Actuator Power for Random Disturbance, 1-lb RMS

type	#	#1	#2	#3	#4	#5	#6	#7	#8	Power
Open	1	0.295	0.274	0.198	0.261	0.261	0.176	0.168	0.178	—
DOFB	2	0.124	0.122	0.092	0.125	0.141	0.135	0.064	0.080	0.659
COFB	3	0.130	0.125	0.105	0.130	0.136	0.134	0.077	0.100	0.596
OD	4	0.137	0.136	0.123	0.124	0.142	0.123	0.088	0.138	0.408
CCS	5	0.144	0.136	0.101	0.164	0.149	0.146	0.076	0.106	0.503

Table 6.3: Simulated RMS Sensor Velocities and Actuator Power for Random Disturbance, 1-lb RMS

type	#	#1	#2	#3	#4	#5	#6	#7	#8	Power
Open	1	0.259	0.268	0.146	0.260	0.264	0.139	0.153	0.156	—
DOFB	2	0.138	0.126	0.092	0.126	0.141	0.092	0.077	0.080	0.776
COFB	3	0.139	0.127	0.093	0.129	0.144	0.098	0.075	0.079	0.729
OD	4	0.154	0.144	0.102	0.143	0.155	0.100	0.101	0.103	0.517
CCS	5	0.154	0.150	0.099	0.157	0.163	0.101	0.090	0.098	0.543

performance level a controller will achieve. The damping ratios for each controller were calculated using the 42 state model, including the first 11 modes as well as the 10 actuator modes. This was done in continuous-time, with no delay in the feedback loop.

For each type of controller listed in Section 5.2, several controllers were designed at various times throughout the project with various weighting matrices. Every type of controller designed has been implemented in the software, however, only some of them are presented here.

Controllers were designed using the pointing requirement derived in Section 5.2. The relative weighting of the Q and R matrices was adjusted so the damping in the first mode was about 17%. This represents the most damping that could be added to the truss using output feedback with 10% actuator damping. If the relative weighting on the matrix R is reduced any further, the damping remains about the same and instability is approached as discussed in Section 5.2.2. The actuator modes become very lightly damped which increases the cost faster than adding damping to the structural modes decreases the cost. The closed loop damping ratios for these controllers are shown in Table 6.4. The actuator damping column is the lowest damping of all the actuator modes.

Table 6.4: Simulation Closed Loop Damping Ratios (in %) — Minimized For Pointing Requirement, 10% Actuator Damping

Control Type	1st x	1st y	1st T	2nd x	2nd y	3rd x	3rd y	2nd T	Actuators
OL	1.70	1.70	0.42	0.42	0.42	0.34	0.34	0.34	9.10
DOFB	11.49	11.65	6.10	2.34	2.35	0.91	0.91	2.18	3.35
COFB	10.12	10.38	6.78	2.62	2.56	0.65	0.65	2.40	4.07
OD	10.12	10.38	6.78	2.62	2.56	0.65	0.65	2.40	4.07
CCS	10.12	10.38	6.78	2.62	2.56	0.65	0.65	2.40	4.07

Table 6.5: Simulation Closed Loop Damping Ratios — Minimized For Pointing Requirement, 50% Actuator Damping

Control Type	#	1st x	1st y	1st T	2nd x	2nd y	3rd x	3rd y	2nd T	Actuators
OL	1	6.32	6.32	1.08	1.07	1.07	0.67	0.68	0.69	49.35
DOFB	2	16.16	17.13	19.73	4.88	5.08	1.61	1.68	6.56	50.17
COFB	3	17.24	18.59	16.34	6.20	6.79	1.36	1.48	5.39	47.40
OD	4	18.09	18.13	17.13	2.72	2.72	1.23	1.23	5.76	48.91
CCS	5	14.76	17.95	8.12	3.03	3.58	1.28	1.49	3.03	47.89

A second group of controllers were designed which were intended to illustrate how additional damping could be achieved by increasing the natural damping of the actuator modes. Damping can be added to the actuators by increasing the gains of the LVT feedback loops, which are adjusted by a 10 turn calibrated dial of the front panel of the actuator power drivers. An actuator damping of 50% was chosen. The open loop structural damping with 50% actuator damping changes, since there is coupling between the actuator modes and the structural modes. As mentioned above, the damping ratios of the closed-loop actuator modes decrease as the damping ratios of the structural modes increase. By adding more damping to the open-loop actuator modes, more damping may be added to structural modes of the closed-loop system. The closed loop damping ratios for these controllers are shown in Table 6.5 and may be compared with the open loop damping ratios.

Attempts were made to estimate the experimental damping ratios from the transfer function data using a circle-fit damping estimation technique. Unfortunately, the results were very sporadic and varied tremendously for various data "windows" to which circles were fitted. For example, when estimating the damping of the first bending mode for the DOFB controller, two separate attempts from the same transfer function data (different windows) yielded estimates of 12% and 36% damping. These variations between data windows may be related to how well the circles fit the data. With such wide variation in the results, these experimental damping estimation attempts were abandoned.

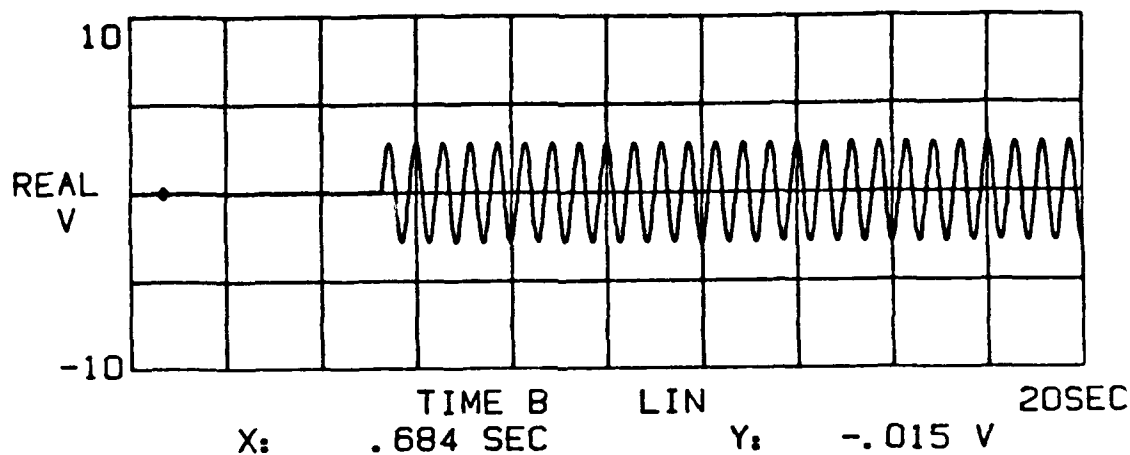


Figure 6.16: Typical Sinusoidal Disturbance Signal, 0.2-lb RMS 1.75 Hz

6.6 Time Responses

Time responses are well suited for examining the transient responses of the truss. Initial responses were taken by applying the disturbance and control simultaneously. Decay responses were obtained by disturbing the controlled truss for 30 seconds, then removing the disturbance while the control remained on.

The damping of each mode can be examined by applying a single frequency sinusoid disturbance to the truss. Figure 6.16 shows the 1.75 Hz signal used in the initial responses (Figures 6.17 and 6.18) and the decay responses (Figures 6.19 through 6.20). The time constants of the exponential rising and falling envelopes do not appear to be the same. The reason for this is not known. It may involve friction in the actuator. Note that these responses were generated with 10% actuator damping and controllers designed for that configuration. Therefore, the attenuation and damping ratios do not match with the discussions from previous sections where actuator damping ratios were 50%.

6.7 H_∞ Gap Metric Design Results

The H_∞ Gap Metric Design experiments addressed different issues than the LQR-based techniques as discussed in Sections 2.3 and 5.2.2. Therefore, we present these results separately.¹

Table 6.6 shows the damping ratios of the controlled modes for the controllers designed

¹This portion of work was performed by Scott Buddie.

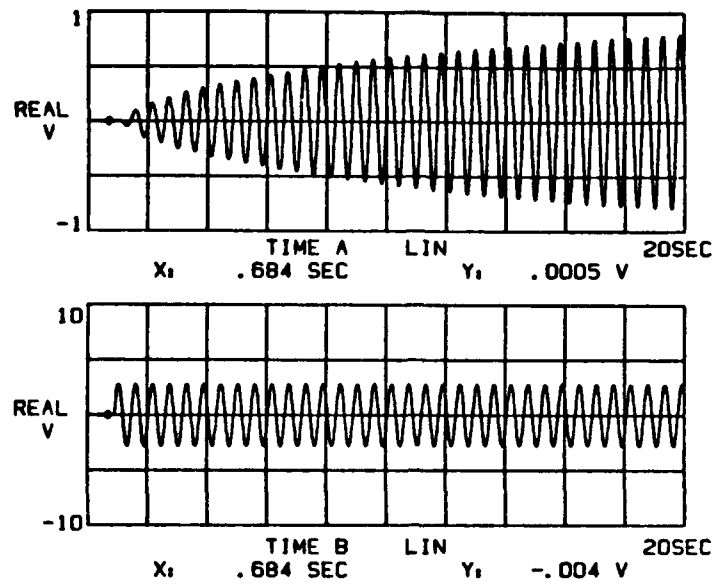


Figure 6.17: Initial Response - Open Loop, Velocity #1, 0.2-lb RMS 1.75-Hz Source 1

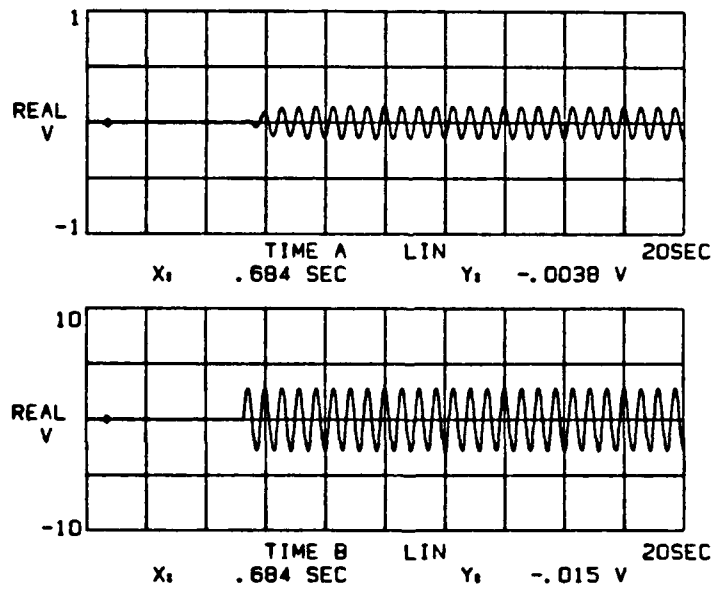


Figure 6.18: Initial Response - DOFB #2, Velocity #1, 0.2-lb RMS 1.75-Hz Source 1

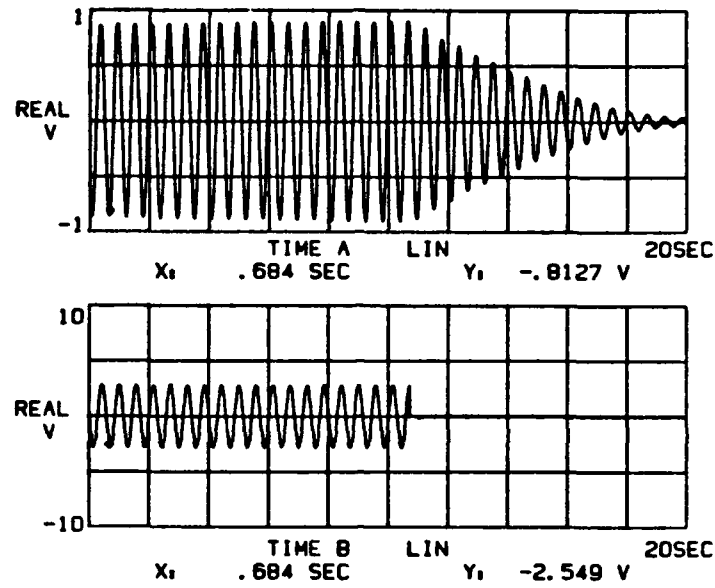


Figure 6.19: Decay Response - Open Loop, Velocity #1, 0.2-lb RMS 1.75-Hz Source 1

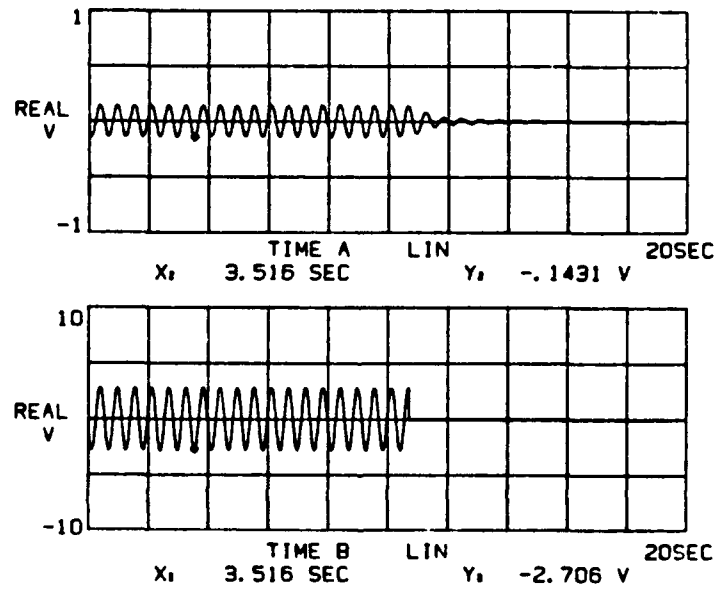


Figure 6.20: Decay Response - DOFB #2, Velocity #1, 0.2-lb RMS 1.75-Hz Source 1

Table 6.6: Damping Ratios of Structural Modes, b_{opt} , and Weighted Gaps for H_∞ Designs

Mode Frequency(Hz)	k	% Damping					b_{opt}	Weighted Gap
		1st x	1st y	1st T	2nd x	2nd y		
		1.66	1.66	6.64	8.45	8.44		
Open Loop	—	6.3	6.3	1.1	1.1	1.1	—	—
Constant	0.333	16.14	16.14	2.78	1.78	1.78	0.7756	0.7761
Weight	0.4	18.93	18.93	3.26	1.89	1.89	0.745	0.82
	0.5	23.21	23.21	3.99	2.39	2.39	0.709	0.88
First Order	$\frac{0.33(0.1s+45)}{s+45}$	15.80	15.80	2.18	1.42	1.42	0.779	0.40
Weight	$\frac{0.50(0.1s+45)}{s+45}$	22.68	22.68	3.11	1.76	1.76	0.712	0.55
Second Order	$\frac{0.50(0.1s+53)^2}{s^2+26.5s+2809}$	22.04	22.06	6.70	4.60	4.60	0.699	0.370
Weight	$\frac{0.65(0.1s+52)^2}{s^2+26s+2704}$	25.53	25.55	8.55	6.04	6.04	0.655	0.446

with the given weighting function. Only the modes which the controller was designed to control are shown. Table 6.6 also shows the calculated values of b_{opt} and the weighted gap between the design model and the truth model of the plant. From this table, we can see that for the case of constant weighting, the damping ratios of the modes increase, the optimal robustness radius decreases, and the weighted gap between the design model and the truth model increases as the constant value k is increased. At $k = 0.333$, the values of the weighted gap and b_{opt} are just about equal to each other.

In addition, closed loop stability of the truth model is checked and verified. This differs from the case of the optimal controller. The damping values of the controlled modes are similar using the optimal controller; however, the truth model was destabilized. This is why the four-block solution is employed for the constant weighting function as discussed in Section 5.2.2.

By using the first order weighting scheme, the optimal robustness radius increases very slightly and the weighted plant gap decreases by a considerable amount. This agrees with intuition because we are de-emphasizing control at the higher frequencies. Note that the damping ratios have decreased from the constant weighting case in the structural modes as expected. In addition, closed loop stability was checked with the truth model. For these controllers, the truth model was stable.

In the previous designs, a large increase in damping is evident in the first x and y bending modes without much of an increase in the damping of the torsional mode or second bending modes. We would like to increase the damping in the torsional mode by increasing the magnitude of the weighting function in the frequency range around the torsional mode. By introducing the second order weighting functions shown in Table 6.6, the damping of the first torsional mode and the second bending modes increase because they are penalized more heavily. This second order weight is applied

to each I/O pair; thus, introducing a weighting function of eight states. Therefore, the optimal controller algorithm yields a dynamic controller with 29 states. Controller reduction yields a 21-state controller for simulation and implementation. Table 6.6 shows the resulting damping ratios and the values of b_{opt} and the weighted gaps. It should be noted that the damping ratios for the first six uncontrolled modes were also calculated for each controller design. The damping ratios for these uncontrolled modes were essentially the same in the presence of the controllers as they were for the open loop case.

Bode plots for constant weighting with $k = 0.5$ and for the two second order weighting cases are shown in Figures 6.21 to 6.23. These plots show the frequency response of the first sensor output velocity with respect to the disturbance input force. The Bode plots of other sensor outputs are very similar.

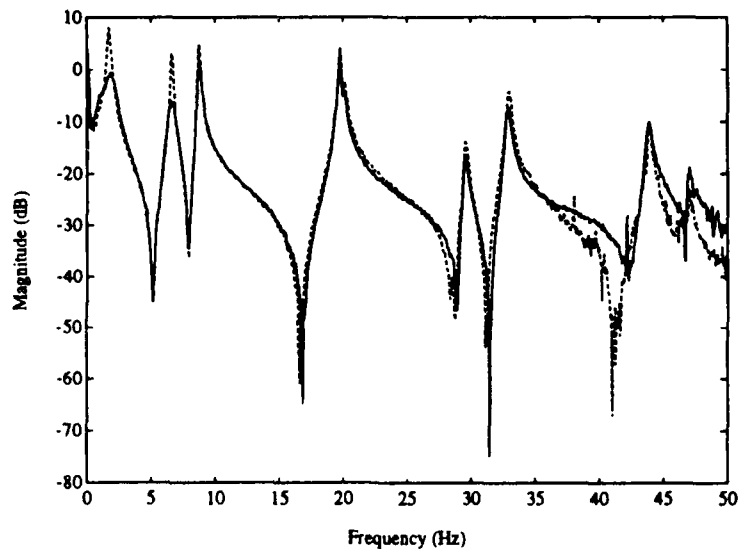


Figure 6.21: Comparison of Experimental Open Loop and Closed Loop Frequency Responses for Constant Weight H_{∞} Design, $k = 0.5$, Velocity #1, Shaker 1-lb RMS Random

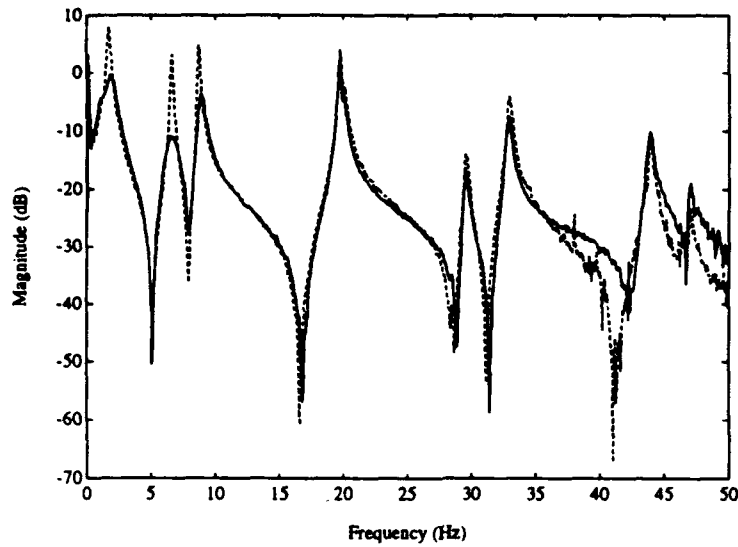


Figure 6.22: Comparison of Experimental Open Loop and Closed Loop Frequency Responses for Second-Order Weight H_∞ Design, $k = \frac{0.50(0.1s+53)^2}{s^2+26.5s+2809}$, Velocity #1, Shaker 1-lb RMS Random

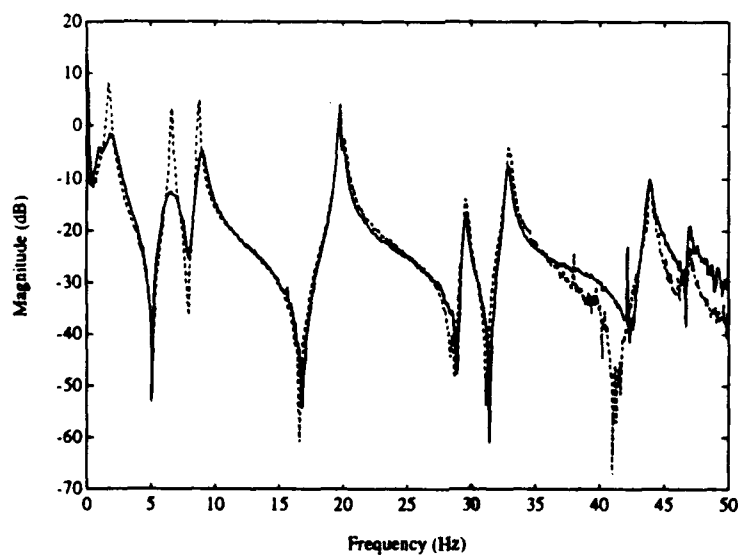


Figure 6.23: Comparison of Experimental Open Loop and Closed Loop Frequency Responses for Second-Order Weight H_∞ Design, $k = \frac{0.65(0.1s+52)^2}{s^2+26s+2704}$, Velocity #1, Shaker 1-lb RMS Random

7. CONCLUSIONS AND RECOMMENDATIONS

This report has been an overview of a fairly extensive study on applying different control techniques on a cantilevered truss structure for active vibration damping. A number of standard issues in the general area of active control of large flexible structures have been considered. These include

1. Modeling
2. Model Reduction
3. Actuator Dynamics
4. Performance Specification
5. Implementation Constraints
 - (a) Computational Delays
 - (b) Actuator Locations
 - (c) Sensor Locations
 - (d) Bandwidth and Authority Limits
 - (e) Unmodeled Dynamics
 - (f) Unknown Parameters
6. Selection of Control Approach
7. Computation of Feedback Gains

Again, as stated above, these are the issues that *any* control designer faced with a large flexible structure will encounter.

On the other hand, the truss structure under consideration was a fairly simple problem for control due to the following facts:

- The behavior of the structure was quite similar to that of a beam.
- Bending vibrations along the both axes and the torsional vibrations were fairly uncoupled.
- The vibrational frequencies (along each axis) were not close enough to create major problems in controller design.

These simplifying aspects notwithstanding, we believe the experimental structure with the multiple actuators and sensors and the computer system provides a very successful testbed and furthermore has provided both the OSU Group and the WL researchers a unique learning experience that is invaluable for work on other large flexible structure problems.

As discussed in the preceding pages a number of different control approaches were analyzed both in simulation and in implementation on the truss. These were:

1. Linear Quadratic Gaussian (LQG)
2. Linear Quadratic Gaussian with Loop Transfer Recovery (LQG/LTR)
3. Centralized Optimal Output Feedback
4. Decentralized Optimal Output Feedback
5. Decentralized Frequency Shaping
6. Overlapping Decomposition (OD)
7. Controlled Component Synthesis (CCS)
8. Maximum Entropy/Optimal Projection (MEOP)
9. Decoupled Optimal Projection (DOP)
10. H_∞ Gap Metric Design

Each one of the approaches implemented was successful in vibration damping. It has to be understood that control design approaches are based on specific criteria such as *minimizing energy*, or *specifying closed loop pole locations* or, *reducing effects of parameter variations*, or *rejecting high frequency effects*, or *adapting behavior to emulate specified dynamics*. Thus one cannot and should not compare these designs on a single criterion, and we have not attempted to do so. Indeed, we have attempted to also consider approaches that have computational and conceptual advantages when they are to be used on more complex structures than the truss configuration here.

REFERENCES

- [1] E. Breitfeller and Ü. Özgüner, "Development of a control orientated model of a fixed-free beam with end-mass," in *Proceedings of 1988 American Control Conference*, (Atlanta, GA), June 1988.
- [2] R. W. Gordon, Ü. Özgüner, and S. Yurkovich, "Large space structures experiments at the flight dynamics laboratory," in *Proceedings of the 3rd NASA/DOD CSI Technology Conference*, (San Diego, CA), January 1989.
- [3] Ü. Özgüner, S. Yurkovich, and R. Gordon, "A case study in control design and implementation," in *Proceedings of the 1989 American Control Conference*, (Pittsburgh, PA), June 1989.
- [4] J. N. Aubrun, "Theory of the control of structures by low-authority controllers," *Journal of Guidance and Control*, vol. 3, pp. 444-451, Sept. 1980.
- [5] N. K. Gupta, "Frequency-shaped cost functionals: Extension of Linear-Quadratic-Gaussian design methods," *Journal of Guidance and Control*, vol. 3, pp. 529-535, November-December 1980.
- [6] Ü. Özgüner and S. Yurkovich, "New directions in decentralized/relegated control of large space structures," Tech. Rep. CRL-1005-Su86-R, Control Research Laboratory, The Ohio State University, June 1986.
- [7] M. Ikeda, D. D. Šiljak, and D. E. White, "Decentralized control with overlapping information sets," *Journal of Optimization Theory and Applications*, vol. 34, pp. 279-310, June 1981.
- [8] M. Ikeda, D. D. Šiljak, and D. E. White, "An inclusion principle for dynamic systems," *IEEE Transactions on Automatic Control*, vol. AC-29, pp. 244-249, March 1984.
- [9] M. Ikeda and D. D. Šiljak, "Overlapping decentralized control with input, state, and output inclusion," *Control Theory and Advanced Technology*, vol. 2, pp. 155-172, June 1986.
- [10] D. D. Šiljak, *Decentralized Control of Complex Systems*. New York: Academic Press, 1991.
- [11] A. İftar and Ü. Özgüner, "Local LQG/LTR controller design for decentralized systems," *IEEE Transactions on Automatic Control*, vol. AC-32, pp. 926-930, October 1987.

- [12] K. D. Young, "A distributed finite element modeling and control approach for large flexible structures," in *AIAA Guidance, Navigation and Control Conference*, (Minneapolis, MN), pp. 253-263, August 1988.
- [13] K. D. Young, "Distributed finite-element modeling and control approach for large flexible structures," *Journal of Guidance, Control, and Dynamics*, vol. 13, no. 4, pp. 703-713, 1990.
- [14] A. İftar and Ü. Özgüner, "Closed-loop balanced realizations in the analysis of suboptimality and stability of decentralized systems," in *Proceedings of the IEEE Conference on Decision and Control*, (Las Vegas, NV), pp. 143-148, 1984.
- [15] D. Sparks, Jr., J. N. Juang, and G. Klose, "A survey of experiments and experimental facilities for control of flexible structures," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, pp. 1176-1185, Aug. 1989.
- [16] J. L. Junkins, ed., *Mechanics and Control of Large Flexible Structures*, vol. 129 of *Progress in Astronautics and Aeronautics*. Washington, DC: American Institute of Aeronautics and Astronautics, 1990.
- [17] R. Strunce, P. Motyka, B. Schley, J. Keat, J. Marggraff, R. O'Donnell, B. Podgorski, and J. Turner, "An investigation of enabling technologies for large precision space structures, vol. iii—a survey of enabling technologies," Tech. Rep. AFRPL-TR-82-074, Air Force Rocket Propulsion Laboratory, September 1982.
- [18] M. A. Floyd and W. E. Vander Velde, "Verification of RCS controller methods for flexible spacecraft," Tech. Rep. AFRPL-TR-84-092 and CDSL-R-1735, Charles Stark Draper Laboratory, 1984.
- [19] L. Meirovich, H. Baruh, R. Montgomery, and J. Williams, "Nonlinear natural control of an experimental beam," *Journal of Guidance and Control*, vol. 7, no. 4, pp. 437-442, 1984.
- [20] Ü. Özgüner, S. Yurkovich, J. Martin, and F. Al-Abbass, "Decentralized control experiments on NASA's flexible grid," in *Proceedings of the 1986 American Control Conference*, vol. 2, pp. 1045-1051, 1986.
- [21] D. Ghose and R. Montgomery, "Problems associated with reaction mass actuators used in conjunction with LQG control on the mini-mast," in *Proceedings of the Second NASA/DoD Control Structures Interaction Technology Conference*, (Wright-Patterson AFB, Ohio), Air Force Wright Aeronautical Laboratory, November 1987. AFWAL-TR-88-3052.

- [22] M. Mercadal and W. E. Vander Velde, "Experimental failure detection results using the SCOLE facility," in *Proceedings of the Second NASA/DoD Control Structures Interaction Technology Conference*, (Wright-Patterson AFB, Ohio), Air Force Wright Aeronautical Laboratory, November 1987. AFWAL-TR-88-3052.
- [23] W. K. Belvin, K. B. Elliott, A. M. Bruner, J. L. Sulla, and J. Bailey, "The LaRC CSI Phase-O Evolutionary Model Testbed: Design and experimental results," in *Proceedings of the Fourth NASA/DoD Control Structures Interaction Technology Conference*, November 1990.
- [24] A. M. Bruner, W. K. Belvin, L. G. Horta, and J. N. Juang, "Active vibration absorber for the CSI Evolutionary Model: Design and experimental results," in *Proceedings of the AIAA 32nd Structures, Structural Dynamics, and Materials Conference*, pp. 2928-2937, April 1991.
- [25] D. B. Schaechter and D. B. Eldred, "Experimental demonstration of the control of flexible structures," *Journal of Guidance, Control, and Dynamics*, vol. 7, no. 5, pp. 527-534, 1984.
- [26] J. L. Fanson and J. C. Chen, "Structural control by use of piezoelectric active members," in *Proceedings of the First NASA/DoD Control Structures Interaction Technology Conference*, November 1986. NASA-CP-2447, Pt. II.
- [27] H. C. Vivian *et al.*, "Flexible structure control laboratory development and technology demonstration," Tech. Rep. 88-29, Jet Propulsion Laboratory, Oct. 1987. Final Report prepared for USAF-AFAL/NASA-OAST.
- [28] J. L. Fanson, G. Blackwood, and C. Chu, "Experimental evaluation of active member control of precision structures," in *Proceedings of the Third NASA/DoD Control Structures Interaction Technology Conference*, January 29-February 2 1989. NASA-CP-3041.
- [29] A. Das, T. J. Strange, Schlaegel, and J. M. Ward, "Experiment in modeling and parameter estimation of flexible structures," in *Proceedings of the Second NASA/DoD Control Structures Interaction Technology Conference*, (Wright-Patterson AFB, Ohio), Air Force Wright Aeronautical Laboratory, November 1987. AFWAL-TR-88-3052.
- [30] R. Quatararo and J. Harris, "ASTREX-A facility for integrated structures and control research," in *Proceedings of the Second NASA/DoD Control Structures Interaction Technology Conference*, (Wright-Patterson AFB, Ohio), Air Force Wright Aeronautical Laboratory, November 1987. AFWAL-TR-88-3052.

- [31] G. A. Norris, "Initial operational capability of the ASTREX Space Structures Testbed," in *Proceedings of the Third NASA/DoD Control Structures Interaction Technology Conference*, pp. 507-522, January 29-February 2 1989. NASA-CP-3041.
- [32] H. B. Waites and V. L. Jones, "Cost effective development of a national test bed," in *Proceedings of the Second NASA/DoD Control Structures Interaction Technology Conference*, (Wright-Patterson AFB, Ohio), Air Force Wright Aeronautical Laboratory, November 1987. AFWAL-TR-88-3052.
- [33] R. D. Irwin, V. L. Jones, S. A. Rice, D. K. Tollison, and S. M. Seltzer, "Active Control Technique Evaluation for Spacecraft," Tech. Rep. AFWAL-TR-88-3038, Air Force Wright Aeronautical Laboratories, June 1988.
- [34] E. Collins, D. J. Phillips, and D. C. Hyland, "Design and implementation of robust decentralized control laws for the ACES structure at Marshall Space Flight Center," Tech. Rep. CR-4310, NASA, July 1990.
- [35] J. Martin, Ü. Özgüner, and S. Yurkovich, "An active vibration-damper for flexible structures," in *17th Annual Pittsburgh Conference on Modeling and Simulation*, (Pittsburgh, PA), April 1986.
- [36] Ü. Özgüner, S. Yurkovich, J. Martin, and P. Kotnik, "A laboratory facility for flexible structure control experiments," *IEEE Control Systems Magazine*, vol. 8, August 1988.
- [37] E. Barbieri and Ü. Özgüner, "Unconstrained and constrained mode expansions for a flexible slewing link," *Journal of Dynamic Systems, Measurement, and Control*, vol. 111, pp. 416-421, December 1988.
- [38] S. Yurkovich and A. Tzes, "Experiments in identification and control of flexible-link manipulators," *IEEE Control Systems Magazine*, vol. 10, pp. 41-47, February 1990.
- [39] S. Yurkovich and Ü. Özgüner, "Recent developments in the OSU flexible structure control laboratory," in *Seventh VPI&SU Symposium on Dynamics and Control of Large Structures*, (Blacksburg, VA), May 1989.
- [40] S. Yurkovich, Ü. Özgüner, and K. Ossman, "Control and identification experiments for the NASA LaRC SCOPE program," tech. rep., NASA, Hampton, VA, 1988. Project Report for NASA Grant NAG-1-720.
- [41] K. Ossman, S. Yurkovich, and Ü. Özgüner, "Adaptive control techniques for the SCOPE configuration," in *Sixth VPI&SU Symposium on Dynamics and Control of Large Structures*, (Blacksburg, VA), June 1987.

- [42] Ü. Özgüner, K. Ossman, J. Donne, M. Boesch, and P. C. Wong, "Decentralized control applied to control of large flexible space structures," final report, Jet Propulsion Laboratory, Pasadena, CA, July 1990.
- [43] R. W. Gordon, "The 12 meter truss active control experiment design, analysis, & open loop testing," Wright Laboratory Technical Report WL-TR-92-3012, Air Force Wright Laboratory, Wright-Patterson AFB, OH, February 1991.
- [44] P. Dix, Ü. Özgüner, and R. W. Gordon, "Decentralized control experiments on a truss structure," in *Proceedings of the 29th IEEE Conference on Decision and Control*, (Honolulu, Hawaii), December 1990.
- [45] M. Balas, "Trends in large space structure control theory: Fondest hopes, wildest dreams," *IEEE Transactions on Automatic Control*, vol. ac-27, no. 3, pp. 522-535, 1982.
- [46] B. Moore, "Principal component analysis in linear systems: Controllability, observability and model reduction," *IEEE Transactions on Automatic Control*, vol. 26, pp. 17-31, 1981.
- [47] E. D. O. Anderson and J. B. Moore, *Linear Optimal Control*. Englewood, New Jersey: Prentice-Hall, 1971.
- [48] F. L. Lewis, *Optimal Control*. New York: John Wiley & Sons, 1986.
- [49] W. S. Levine and M. Athans, "On the determination of the optimal constant output-feedback gains for linear multivariable systems," *IEEE Transactions on Automatic Control*, vol. AC-15, pp. 44-48, February 1970.
- [50] B. Friedland, *Control System Design*. New York: McGraw-Hill, 1986.
- [51] J. C. Doyle and G. Stein, "Multivariable feedback design: Concepts for a classical/modern synthesis," *IEEE Transactions on Automatic Control*, vol. AC-26, pp. 4-16, February 1981.
- [52] G. Stein and M. Athans, "The LQG/LTR procedure for multivariable feedback control design," *IEEE Transactions on Automatic Control*, vol. AC-32, pp. 105-114, February 1987.
- [53] E. J. Davison and I. J. Ferguson, "The design of controllers for the multivariable robust servomechanism problem using parameter optimization methods," *IEEE Transactions on Automatic Control*, vol. AC-26, pp. 93-110, Feb. 1981.

- [54] E. J. Davison and T. Chang, "The design of decentralized controllers for the robust servomechanism problem using parameter optimization methods— Some case studies," in *Proceedings of the IEEE Conference on Decision and Control*, (Orlando, Florida), pp. 548–553, Dec. 1982.
- [55] A. İftar and Ü. Özgüner, "Examples of decentralized servocompensators for flexible structures," Tech. Rep. CRL-1007-Su86-R, Control Research Laboratory, The Ohio State University, 1986.
- [56] Ü. Özgüner and S. Yurkovich, "Decentralized frequency shaping and model sensitivities for optimal control of large space structures," in *Proceedings of the 10th IFAC World Congress*, (Munich, West Germany), pp. 6.47–6.52, July 1987.
- [57] G. Guardabassi, A. Locatelli, C. Maffezzoni, and N. Schiavoni, "Computer-aided design of structurally constrained multivariable regulators," *IEE Proceedings*, vol. 130, pp. 155–172, July 1983. Parts I and II.
- [58] F. Khorrami, S. Tien, and Ü. Özgüner, "DOLORES: a software package for analysis and design of optimal decentralized control," in *Proceedings of the 40th National Aerospace and Electronics Conference*, (Dayton, OH), May 1988.
- [59] F. Khorrami and Ü. Özgüner, "Frequency-shaped cost functionals for decentralized systems," in *Proceedings of the IEEE Conference on Decision and Control*, (Austin, TX), December 1988.
- [60] B. D. O. Anderson, J. B. Moore, and D. L. Mingori, "Relations between frequency-dependent control and state weighting in LQG problems," *Optimal Control Applications and Methods*, vol. 8, pp. 109–127, 1987.
- [61] S. H. Wang and E. J. Davison, "On the stabilization of decentralized control systems," *IEEE Transactions on Automatic Control*, vol. AC-18, pp. 473–478, 1973.
- [62] Ü. Özgüner and W. R. Perkins, "Structural properties of large-scale composite systems," in *Large-Scale Dynamical Systems* (R. Seaks, ed.), (N. Hollywood, CA), Point Lobos Press, 1975.
- [63] T. Williams, "Closed-form grammians and model reduction for flexible space structures," *IEEE Transactions on Automatic Control*, vol. AC-35, pp. 379–382, Mar. 1990.
- [64] W. C. Hurty, "Vibration of structural systems by component mode synthesis," *ASCE Journal of the Engineering Mechanics Division*, vol. 85, pp. 51–69, 1960.

- [65] W. C. Hurty, "Dynamic analysis of structural systems using component mode synthesis," *AIAA Journal*, vol. 3, no. 4, pp. 678-685, 1965.
- [66] J. T. Spanos and W. S. Tsuha, "Selection of component modes for flexible multi-body simulation," *Journal of Guidance, Control, and Dynamics*, vol. 14, no. 2, pp. 278-286, 1991.
- [67] R. R. Craig, Jr., "Methods of component mode synthesis," *Shock and Vibration Digest*, vol. 9, pp. 3-10, 1977.
- [68] R. R. Craig, Jr., *Structural Dynamics: An Introduction to Computer Methods*. New York: John Wiley & Sons, 1981.
- [69] R. R. Craig, Jr. and M. C. C. Bampton, "Coupling of substructures for dynamic analysis," *AIAA Journal*, vol. 6, no. 7, pp. 1313-1319, 1968.
- [70] R. H. MacNeal, "A hybrid method of component mode synthesis," *Computers and Structures*, vol. 1, no. 4, pp. 581-601, 1971.
- [71] S. Rubin, "Improved component-mode representation for structural dynamic analysis," *AIAA Journal*, vol. 13, no. 8, pp. 995-1006, 1975.
- [72] W. A. Benfield and R. F. Hrudu, "Vibration analysis of structures by component mode substitution," *AIAA Journal*, vol. 9, no. 7, pp. 1255-1261, 1971.
- [73] D. C. Hyland, "Optimal regulation of structural systems with uncertain parameters," Tech. Rep. 551, Massachusetts Institute of Technology, Lincoln Laboratory, February 1981.
- [74] D. C. Hyland, "Minimum information stochastic modelling of linear systems with a class of parameter uncertainties," in *Proceedings of the American Control Conference*, (Arlington, Virginia), pp. 620-627, June 1982.
- [75] D. C. Hyland, "Maximum entropy stochastic approach to control design for uncertain structural systems," in *Proceedings of the American Control Conference*, (Arlington, Virginia), pp. 680-688, June 1982.
- [76] S. Shokoohi, L. M. Silverman, and V. Dooren, "Linear time-variable systems: Balancing and model reduction," *IEEE Transactions on Automatic Control*, vol. AC-28, pp. 810-822, August 1983.
- [77] A. Yousuff and R. E. Skelton, "Controller reduction by component cost analysis," *IEEE Transactions on Automatic Control*, vol. AC-29, pp. 520-530, June 1984.

- [78] R. N. Mishra and D. A. Wilson, "A new algorithm for optimal reduction of multivariable systems," *International Journal of Control*, vol. 31, no. 3, pp. 443–466, 1980.
- [79] R. E. Skelton, "Cost decomposition of linear systems with application to model reduction," *International Journal of Control*, vol. 32, no. 6, pp. 1031–1055, 1980.
- [80] Y. Liu and B. O. Anderson, "Controller reduction via stable factorization and balancing," *International Journal of Control*, vol. 44, no. 2, pp. 507–531, 1986.
- [81] S. W. Greeley and D. C. Hyland, "Reduced-order compensation: Lqg reduction versus optimal projection," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, (Monterey, California), pp. 605–616, August 1987.
- [82] D. C. Hyland and D. S. Bernstein, "The optimal projection equations for fixed-order dynamic compensation," *IEEE Transactions on Automatic Control*, vol. AC-29, pp. 1034–1037, November 1984.
- [83] D. S. Bernstein and D. C. Hyland, "The Optimal Projection/Maximum Entropy approach to designing low-order, robust controllers for flexible structures," in *Proceedings of the IEEE Conference on Decision and Control*, (Ft. Lauderdale, Florida), pp. 745–752, December 1985.
- [84] S. A. Buddie, "Evaluation of the weighted gap metric for optimal control of flexible structures," Master's thesis, The Ohio State University, Columbus, OH, June 1991.
- [85] T. Georgiou, "On the computation of the gap metric," *Systems and Control Letters*, vol. 11, pp. 253–257, 1988.
- [86] T. Georgiou and M. Smith, "Robust control of feedback systems with combined plant-controller uncertainty," in *Proceedings of The American Control Conference*, 1990.
- [87] T. Georgiou and M. Smith, "Robust stabilization in the gap metric: Controller design for distributed plants," in *Proceedings of The American Control Conference*, 1990.
- [88] T. Georgiou and M. Smith, *Robust Stabilization in the Gap Metric*. Birkhauser, 1990. Book Chapter to appear.
- [89] M. J. Englehart and M. C. Smith, "A four-block problem for h_∞ design: Properties and applications," *Automatica*, vol. 27, pp. 811–818, 1991.

- [90] K. Glover, "All optimal hankel-norm approximations of linear multivariable systems and their L^∞ error bounds," *Int. J. Control*, vol. 39, no. 6, pp. 1115–1193, 1984.
- [91] K. Glover and D. McFarlane, "Robust stabilization of normalized coprime factor plant descriptions with H^∞ -bounded uncertainty," *IEEE Transactions on Automatic Control*, vol. 34, pp. 821–830, August 1989.
- [92] D. C. McFarlane and K. Glover, *Robust Controller Design Using Normalized Coprime Factor Plant Descriptions*, vol. 138 of *Lecture Notes in Control and Information Sciences*. New York: Springer-Verlag, 1989.
- [93] M. Safonov and R. Chiang, "A schur method for balanced model reduction," in *Proceedings of The American Control Conference*, pp. 1036–1040, 1988.
- [94] The MathWorks, Inc., Natick, MA, *MATLAB User's Guide*, January 1990.
- [95] S. P. Harbison and G. L. J. Steele, *C: A Reference Manual*. Englewood Cliffs, NJ: Prentice-Hall, 1984.
- [96] A. Cagle and Ü. Özgüner, "Optimal decentralized feedback control for a truss structure," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, (Boston, MA), August 1989.
- [97] F. Khorrami, S. Tien, and Ü. Özgüner, *DOLORES Users Guide*. Department of Electrical Engineering: Ohio State University, 1988.
- [98] A. Cagle, "Decentralized controller designs for a large truss structure," Master's thesis. The Ohio State University, Columbus, OH, 1989.

A. Sample Run of Control/Simulation Program

Integrated accelerometer or LVT feedback (a/l) ? a

Control of 12m Truss Response
Ohio State University

Minimum sampling times are given in parenthesis
Active controllers must be run at this sampling time

Actuator damping at 50%

1) Open Loop Response

Designs Minimized for Pointing Requirement - Cheap Control

- 2) Decentralized Optimal Output Feedback (2ms)
- 3) Centralized Optimal Output Feedback (2ms)
- 4) Overlapping Decomposition (2ms)
- 5) Controlled Component Synthesis (2ms)
- 6) LVT Decentralized Optimal Output Feedback (2ms)
- 7) LVT Centralized Optimal Output Feedback (2ms)
- 8) LVT Overlapping Decomposition (2ms)

Designs Minimized for Pointing Requirement - Medium Damping

- 10) Decentralized Optimal Output Feedback (2ms)
- 11) Centralized Optimal Output Feedback (2ms)
- 12) Overlapping Decomposition (2ms)
- 13) LVT Decentralized Optimal Output Feedback (2ms)
- 14) LVT Centralized Optimal Output Feedback (2ms)

Designs Minimized for Pointing Requirement - Light Damping

- 15) Decentralized Optimal Output Feedback (2ms)
- 16) Centralized Optimal Output Feedback (2ms)
- 17) Overlapping Decomposition (2ms)
- 18) Controlled Component Synthesis (2ms)

Designs Based on Weighted Gap Metric

- 20) Full Order with Constant Weights (5ms)
- 21) Reduced Order with Constant Weights (5ms)
- 22) First Order Weight - Heavy Damping
- 23) Second Order Weight
- 24) Smith & Georgiou Constant Weight (5ms)
- 25) Smith & Georgiou Second Order Light (5ms)
- 26) Smith & Georgiou Second Order Medium (5ms)
- 27) Smith & Georgiou Second Order Heavy (5ms)

Actuator damping at 10%

30) Open Loop Response

Choose type of control (1-30) 1

Control truss or simulate (c/s) ? s
Enter sampling rate in milliseconds 2
Enter data recording period in milliseconds (multiple of sampling rate) 10

Disturbance Sources

- 1) Disturbance actuator mounted on truss
- 2) Shaker mounted on wall

Choose type of disturbance excitation (1-2) 2

... Loading Model Files
... getting ./matrices/c_42.dat
... getting ./matrices/cd_42.dat
... getting ./matrices/ctravel42.dat
... Loading Controller Files and Discretizing
... Loading Model with 50% Actuator damping
... getting ./matrices/a_42_a50_d2.dat
... getting ./matrices/bsw_42_a50_d2.dat
Enter length of time in seconds for data recording 10
Number of recorded data points is 1000
Is this OK ? (y/n) y
Enter delay in seconds before beginning recording 0
Enter length of time in seconds to control truss 11

Disturbance Signals

- 1) Bandwidth Limited Gaussian White Noise
- 2) Sinusoidal at one frequency
- 3) Sinusoidal Sweep
- 4) Burst of Bandwidth Limited Gaussian White Noise
- 5) Burst of Sinusoidal at one frequency

Which type of disturbance? (1-5) 1

Enter RMS value of disturbance 1

Cutoff is 50Hz

... computing disturbance signal
... getting ./disturbance_filters/fir50_2.dat
corrected mean is -0.000117
corrected std is 1.000000

Enter filename with no extension for data (will be put in ./data/) tmp

Simulation Loop Running

... 5% Complete
... 10% Complete
... 20% Complete
... 50% Complete
... 80% Complete
... uploading .m File
... opening data file

RMS pointing error is 0.019642

RMS Force is 1.010897

